EXHIBIT 9

Copyright © 1982 American Telephone and Telegraph Company
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 61, No. 7, September 1982
Printed in U.S.A.

Description of Fasnet—A Unidirectional Local-Area Communications Network

By J. O. LIMB and C. FLORES*

(Manuscript received January 15, 1982)

Fasnet is an implicit token-passing, local-area network aimed at supporting high data rates and carrying a wide mix of traffic (data, voice, video, and facsimile). Transmission is unidirectional with stations attaching to the medium passively via directional couplers. A link consists of two lines, one to carry traffic in each direction. Unidirectional transmission provides the potential for efficient operation at high data rates, while the passive medium provides the potential for high reliability. We describe the physical configuration and the protocol and give channel utilization for the condition of continuously queued sources. Mechanisms to control the access of various traffic types are described. Finally, the interconnection of multiple Fasnets is studied for one particular configuration, a ring.

I. INTRODUCTION

Local computer networks operating at 1 to 10 Mb/s are being commercially offered and appear to adequately meet current demands for computer communications within the office environment. However, future needs stimulated by both a broader range of services than is now available and changes in system architecture (e.g., the trend towards distributed processing) could increase significantly the demand for digital capacity. For example, one would like to be able to handle video information, voice traffic, and facsimile, as well as computer traffic, in a single digital system. The availability of a cheap, high-capacity communication conduit between computers will itself stimulate increased traffic. For example, processing time can be traded for communication capacity; rather than transmitting a text file and

* Department of Electrical Engineering and Computer Science, University of California at Berkeley.

formatting it at a remote location, one may choose to transmit a formatted version or even a bit map. Thus, while today 10 Mb/s may be regarded as an extremely generous bit rate for a local computer network, 200 Mb/s may become limiting for an integrated communications network.

Carrier-sense multiple access with collision detection (CSMA/CD) is reliable and reasonably efficient even under heavy load for most conditions.[1] Shoch and Hupp[2] show that measurements of channel utilization of an Ethernet[*] yield results that are close to calculations made by Metcalf and Boggs, using a simplified model and assuming that active stations have data continuously queued for transmission. Utilization is reported in Ref. 1: $\eta = \gamma/[\gamma + F(M)]$, where γ is the ratio of packet duration to slot time and $F(M)$ is a slowly varying function of M, the number of active stations. Utilization is plotted as a function of γ in Fig. 1, with M as a parameter. If we assume that slot time is 50 μs,[3] and the transmission rate is 100 Mb/s, then for an average packet length of 1000 bits, $\gamma = 0.2$ and utilization is in the range 7 to 8 percent.[†] Shorter packets, higher transmission rates, or longer slot times would further decrease efficiency. Also note that the above equation does not incorporate source acquisition and synchronization time which, like slot time, is relatively more significant at higher transmission speeds. Thus, it appears that CSMA/CD is not viable for operation at high data rates.[5]

Fasnet is an implicit token-passing protocol developed to efficiently utilize the channel capacity when the ratio of packet duration to the maximum station-to-station propagation time is small (<1). Information flows in only one direction on the medium, unlike the usual CSMA/CD configurations (although see Refs. 5 and 6), but like CSMA/CD the essential passivity of the medium is retained. The access method is closely related to a ring protocol (e.g., see Ref. 7) and may be regarded as a variant of implicit token passing.

Reliability was an important consideration in the design of Fasnet. Consider both the transmission medium and the control electronics. Reliability of the transmission medium may be enhanced by keeping active electronics in the medium to a minimum. Bus architectures such as Ethernet have occasional repeaters, depending on the length of the signal path. Cable-TV (CATV) type architectures have periodic line amplifiers whose spacing is determined by the number of stations (taps), as well as by cable attenuation. Ring architectures usually have most electronics in the signal path; digital
provided at each station.[7] Turning to control,
tend to favor distributed control. An alternat
system is to permit some stations to perfo
have these functions assumable by any stati
ever, this can result in a large cost penalty. A
have the function performed on a server ba
provide the service to all other stations). Th
bles that employed in CATV, and control is p
some functions assumable by all stations.

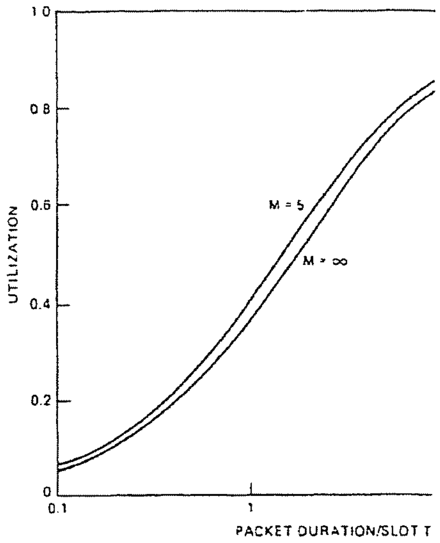


Fig. 1—Plot of efficiency as a function of the ratio o
Parameter is the number of simultaneous users. Efficien
of users goes from 5 to ∞.

There is considerable attraction in having
all forms of traffic in the environment. Indee
regarded as an extension into the local envir
Systems Digital Network (ISDN) that is being
the long distance and local loop environments
An integrated transmission system simplifie
services that utilize different types of traff
annotated electronic mail[9] and interactive us
An integrated transmission system also pro
reduce overall transmission needs by taking
mentary nature of some types of traffic; for

[*] Ethernet is a trademark of Xerox Corporation.

[†] Since Ethernet requires $\gamma \geq 1.0$ for collisions to be reliably detected, utilization for values of $\gamma < 1.0$ were obtained by multiplying the utilization for $\gamma = 1$ by the value of γ. This can result in efficiencies much lower than that obtainable by other CSMA protocols, e.g., p-persistent CSMA.[1]

mote location, one may choose to transmit a
ven a bit map. Thus, while today 10 Mb/s may
remely generous bit rate for a local computer
ay become limiting for an integrated communi-

ple access with collision detection (CSMA/CD) is
ly efficient even under heavy load for most
d Hupp[2] show that measurements of channel
net* yield results that are close to calculations
Boggs, using a simplified model and assuming
ave data continuously queued for transmission.
in Ref. 1: $\eta = \gamma/[\gamma + F(M)]$, where γ is the ratio
slot time and $F(M)$ is a slowly varying function
tive stations. Utilization is plotted as a function
is a parameter. If we assume that slot time is 50
on rate is 100 Mb/s, then for an average packet
= 0.2 and utilization is in the range 7 to 8
kets, higher transmission rates, or longer slot
decrease efficiency. Also note that the above
rporate source acquisition and synchronization
time, is relatively more significant at higher
Thus, it appears that CSMA/CD is not viable for
rates.[5]

token-passing protocol developed to efficiently
pacity when the ratio of packet duration to the
tation propagation time is small (<1). Informa-
lirection on the medium, unlike the usual CSMA/
nough see Refs. 5 and 6), but like CSMA/CD the
he medium is retained. The access method is
g protocol (e.g., see Ref. 7) and may be regarded
token passing.

nportant consideration in the design of Fasnet.
nsmission medium and the control electronics.
smission medium may be enhanced by keeping
he medium to a minimum. Bus architectures
occasional repeaters, depending on the length
ble-TV (CATV) type architectures have periodic
pacing is determined by the number of stations
ole attenuation. Ring architectures usually have

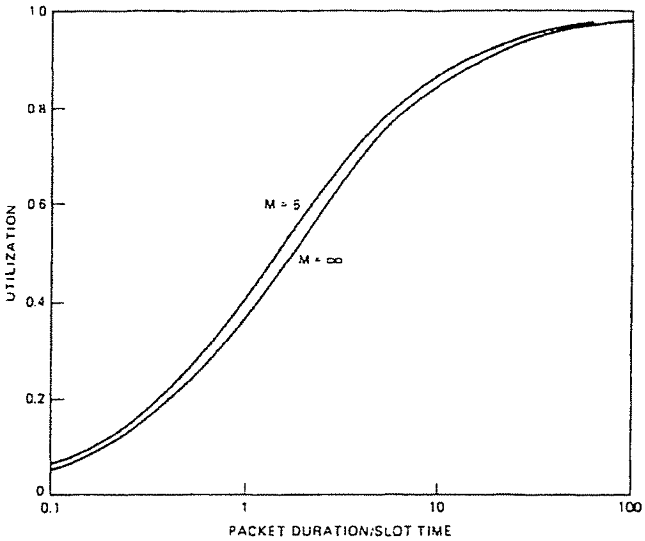


Fig. 1—Plot of efficiency as a function of the ratio of packet duration to slot time. Parameter is the number of simultaneous users. Efficiency changes little as the number of users goes from 5 to ∞.

most electronics in the signal path; digital regeneration is usually provided at each station.[7] Turning to control, reliability considerations tend to favor distributed control. An alternative to a fully distributed system is to permit some stations to perform unique functions but have these functions assumable by any station on the network; however, this can result in a large cost penalty. A further alternative is to have the function performed on a server basis (two or more stations provide the service to all other stations). The Fasnet medium resembles that employed in CATV, and control is primarily distributed with some functions assumable by all stations.

There is considerable attraction in having a single system to handle all forms of traffic in the environment. Indeed, such a system may be regarded as an extension into the local environment of the Integrated Systems Digital Network (ISDN) that is being so vigorously pursued in the long distance and local loop environments by the common carriers.[8] An integrated transmission system simplifies the implementation of services that utilize different types of traffic. Examples are voice-annotated electronic mail[9] and interactive use of voice and facsimile.[10] An integrated transmission system also provides the opportunity to reduce overall transmission needs by taking advantage of the complementary nature of some types of traffic; for example, most electronic

of Xerox Corporation.
$\gamma \geq 1.0$ for collisions to be reliably detected, utilization for
ied by multiplying the utilization for $\gamma = 1$ by the value of
encies much lower than that obtainable by other CSMA
SMA.[1]

mail can be deferred until after the voice busy hour. Further, one would anticipate cost reductions in having one integrated system over a number of separate systems.

While design of an integrated system offers opportunities, it also presents the need for compromises and trade-offs. Consider terminal costs and transmission efficiency. If a system is to provide economical interconnection for telephones and terminals, it must permit construction of an interface that is cost-effective relative to alternative solutions. This may mean that some interfaces have to be tailored to the specific application to make them competitive.

Virtually any type of traffic should be able to exploit the channel efficiently. For example, environments that generate a large number of short messages (e.g., computer terminal traffic), as well as environments that generate a preponderance of long messages (e.g., file transfers), should be able to operate efficiently. This requires that there be a minimum of structure at the lowest common level of the service. For example, a packet structure which mandated a source-address field, while useful for computer traffic, may be unnecessary overhead for a voice channel where call set-up would establish the identity of the source.

The description of Fasnet starts in Section II with the physical loop; the access protocol is described in Section III. The performance of the basic system is given in Section IV, followed by a discussion of some of the system design issues (in particular, the synchronization and signaling procedures) in Section V. Section VI describes variations of the basic system, including methods for improving efficiency, particularly when the number of users is small. Section VII describes mechanisms to support the efficient management and control of mixtures of different traffic types. Section VIII describes the interconnection of Fasnets, with consideration of the impact of the topology on throughput and ability to handle localized sources of traffic. Section IX summarizes the paper.

II. PHYSICAL CONFIGURATION

The basic link, as shown in Fig. 2, consists of two lines. One line passes all stations carrying traffic in one direction and the other line passes all stations carrying traffic in the other direction. For line A, station S_1 is referred to as the head station and S_n the end station. For line B the assignment is reversed. Together the two lines provide a connection between any pair of stations attached to the link. While the lines may be either twisted pair, coaxial cable, or light fibers, we will be primarily concerned with a coaxial cable implementation. Each station makes two connections to each line. A read tap precedes a

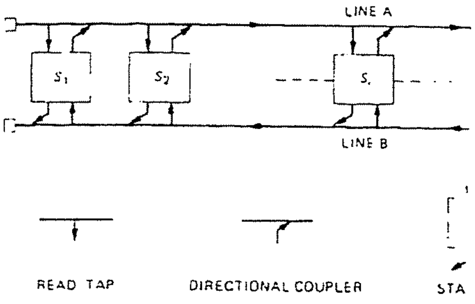


Fig. 2—Physical configuration of a

passive directional coupler used for writing.
tional coupler is such that very little ener
direction on the line so that the signal read
from the read tap will be unaffected by the si
line via the directional coupler. A station wr
energy to the signal already existing on the
fields of the header, the protocol ensures t
time writes on the line. Thus, once a signal is
removed or changed by any station. This ha
the line code that is selected (Section 5.2).

Depending on the length of the line, ampl
and compensate the signal. The technolog
used for CATV systems[11] are directly applicabl
margin required for a high-quality video si
than that required for two- or three-level di

Links may be joined together to form a r
links will be run in pairs of lines, but this is r
advantage of using multiple links is that th
of the network can be increased and reliabi
the use of redundant paths.

An earlier version of Fasnet[12] differs pr
described here in that a link consists of a sing
passes each station twice—on the outbound
inbound or read side. Each station makes thr
a read tap for control purposes, and a directi
side, and a read tap for recovering data on t
advantage of the scheme described here is
approximately twice the traffic of the earlie
is that a station must select the correct line
this will depend on the relative physical loc

until after the voice busy hour. Further, one
reductions in having one integrated system over
systems.
integrated system offers opportunities, it also
compromises and trade-offs. Consider terminal
efficiency. If a system is to provide economical
ephones and terminals, it must permit construc-
hat is cost-effective relative to alternative solu-
that some interfaces have to be tailored to the
make them competitive.
of traffic should be able to exploit the channel
le, environments that generate a large number
., computer terminal traffic), as well as environ-
a preponderance of long messages (e.g., file
able to operate efficiently. This requires that
of structure at the lowest common level of the
a packet structure which mandated a source-
seful for computer traffic, may be unnecessary
channel where call set-up would establish the

asnet starts in Section II with the physical loop;
described in Section III. The performance of the
in Section IV, followed by a discussion of some
issues (in particular, the synchronization and
in Section V. Section VI describes variations of
ding methods for improving efficiency, particu-
r of users is small. Section VII describes mech-
efficient management and control of mixtures of
Section VIII describes the interconnection of
ation of the impact of the topology on through-
andle localized sources of traffic. Section IX

URATION

hown in Fig. 2, consists of two lines. One line
rying traffic in one direction and the other line
rying traffic in the other direction. For line A,
as the head station and S_n the end station. For
is reversed. Together the two lines provide a
ny pair of stations attached to the link. While
r twisted pair, coaxial cable, or light fibers, we
rned with a coaxial cable implementation. Each
nnections to each line. A read tap precedes a

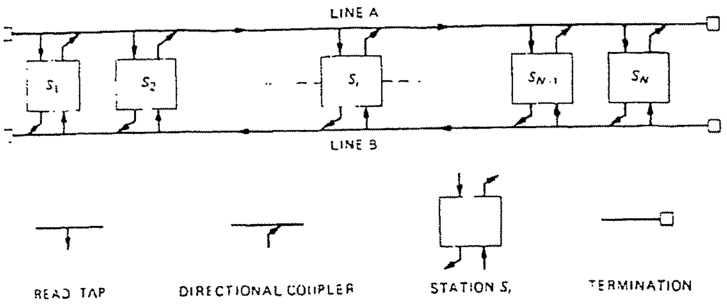


Fig. 2—Physical configuration of a Fasnet link.

passive directional coupler used for writing. The nature of the directional coupler is such that very little energy travels in the reverse direction on the line so that the signal read virtually simultaneously from the read tap will be unaffected by the signal being written on the line via the directional coupler. A station writes on the line by adding energy to the signal already existing on the line. Except for specific fields of the header, the protocol ensures that only one station at a time writes on the line. Thus, once a signal is written on a line, it is not removed or changed by any station. This has certain implications for the line code that is selected (Section 5.2).

Depending on the length of the line, amplifiers are needed to boost and compensate the signal. The technology and design procedures used for CATV systems[11] are directly applicable here, although the noise margin required for a high-quality video signal is somewhat greater than that required for two- or three-level digital transmission.

Links may be joined together to form a network of links. Usually, links will be run in pairs of lines, but this is not always necessary. The advantage of using multiple links is that the traffic-carrying capacity of the network can be increased and reliability may be improved by the use of redundant paths.

An earlier version of Fasnet[12] differs primarily from the system described here in that a link consists of a single unidirectional line that passes each station twice—on the outbound or write side and on the inbound or read side. Each station makes three connections to the line, a read tap for control purposes, and a directional write tap on the write side, and a read tap for recovering data on the read side. The primary advantage of the scheme described here is that the link can carry approximately twice the traffic of the earlier version. A disadvantage is that a station must select the correct line on which to transmit, and this will depend on the relative physical location of the destination.

III. PROTOCOL DESCRIPTION

The data link layer may be divided into two sublayers.[13] One sublayer, the logical link control with which we are less concerned here, provides functions like addressing, windowing, and acknowledgments. The other sublayer, the media access control with which we are more concerned, determines when and how to send information via the physical medium. This is influenced by the media type, the physical configuration, and the technology used.

3.1 Frame format

The frame structure suggested in Ref. 13 and its relation to the data link sublayers is shown in Table I. The information unit is delivered by the network layer. The logical link control appends the source address, the destination address, the link control field for windowing, acknowledgments, and similar functions. We call this unit a packet, and in the work described here we will assume it is of fixed length. The media access control sublayer appends (*i*) the frame check sequence computed on the previous fields for error detection and (*ii*) the access control (AC) field which determines how and when each station may access the physical medium. The main objective in the design of this field is to control access among all active stations in an efficient, reliable, and fair manner. The frame start and frame end delimiters are unnecessary, since the stations are kept in tight bit and frame synchronization (see Section 5.1). The duration of the frame is referred to as a slot.

3.2 Access control

Basic access control for Fasnet is as follows. The head station, S_1, initiates a cycle on line A. After a cycle has been initiated, each active station on the line with packets destined in the right direction is allowed to access the line for one slot. To do this, each station monitors the line. When it senses the line idle, it seizes the line for one slot. It has to wait for a new cycle to be initiated before it attempts to access the line again. The exact manner in which this is done efficiently and fairly is described in the next paragraphs. If
given permission to access the line for an in
this manner, the active stations can access
duration in the order in which they are phys
The operation on line B is identical to that
replacing S_1 as head station.

To describe the operation in more detail, l
set of stations in the order of their physical
2. Let AQ_i and BQ_i be the number of packet
access to lines A and B, respectively.

When the next packet arrives at S_i from th
if destination address $j > i$, then AQ_i is in
if destination address $j < i$, then BQ_i is in

The structure of the AC field is shown in
of the nth frame. The AC field is from t_{fn} to t
to line A in the following manner. Let S_i be
packets each cycle. At t_{fn}, the start of the nth
the START bit of the AC field. The start
START = 1. Because of gate delays in t
propagation delays in the tap cables, the outc
is only known at t_{sn}. This additional time
Fig. 3, is of the order of a few bit times
nanosecond logic. Next, the station may
BUSY bit via the read tap and write BUS
coupler. Again, the outcome of the read oper
after a delay of τ_{dec}. Nonetheless, the write
the BUSY bit if it is already set to 1. The
achieve this is explained in Section 5.2. At t_{bn}
defers until the BUSY bit of the next frame.
accesses the line for the remaining frame du

Station S_i is said to be in one of four state
IDLE —if it has no packets to transmit, i.
WAIT—if it is waiting for the start of cycl

Table I—Protocol and frame structures

	Protocol Structure	Frame Structure
Data link layer	Logical link control sublayer Media access control sublayer	DA/SA/LC/IU FS/AC/DA/SA/LC/IU/FCS/FE
Physical layer	Physical layer signaling	FS/AC/DA/SA/LC/IU/FCS/FE

FS:	Frame starting delimiter	LC:	Link control field
AC:	Access control field	IU:	Information unit from network layer
DA:	Destination address	FCS:	Frame check sequence
SA:	Source address	FE:	Frame ending delimiter

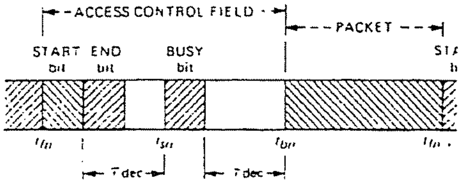


Fig. 3—The frame structure of Fasnet. Each frame
field containing START, END, and BUSY bits; and (i
logical link sublayer.

IPTION

r may be divided into two sublayers.[13] One ink control with which we are less concerned s like addressing, windowing, and acknowledg- ayer, the media access control with which we etermines when and how to send information m. This is influenced by the media type, the and the technology used.

suggested in Ref. 13 and its relation to the data h in Table I. The information unit is delivered The logical link control appends the source n address, the link control field for windowing, similar functions. We call this unit a packet, ibed here we will assume it is of fixed length. trol sublayer appends (*i*) the frame check se- the previous fields for error detection and (*ii*) l) field which determines how and when each physical medium. The main objective in the o control access among all active stations in an fair manner. The frame start and frame end sary, since the stations are kept in tight bit and (see Section 5.1). The duration of the frame is

for Fasnet is as follows. The head station, S_1, A. After a cycle has been initiated, each active ith packets destined in the right direction is ne for one slot. To do this, each station monitors is the line idle, it seizes the line for one slot. It ycle to be initiated before it attempts to access ct manner in which this is done efficiently and

—Protocol and frame structures

icture	Frame Structure
control sublayer	DA/SA/LC/IU
ss control sublayer	FS/AC/DA/SA/LC/IU/FCS/FE
er signaling	FS/AC/DA/SA/LC/IU/FCS/FE

ır LC: Link control field
IU: Information unit from network layer
FCS: Frame check sequence
FE: Frame ending delimiter

fairly is described in the next paragraphs. If a station has priority, it is given permission to access the line for an integral number of slots. In this manner, the active stations can access the line for a specified duration in the order in which they are physically located on the line. The operation on line B is identical to that described above with S_N replacing S_1 as head station.

To describe the operation in more detail, let $\{S_1, S_2, \cdots, S_N\}$ be the set of stations in the order of their physical locations as shown in Fig. 2. Let AQ_i and BQ_i be the number of packets queued at station S_i for access to lines A and B, respectively.

When the next packet arrives at S_i from the Network layer interface,
if destination address $j > i$, then AQ_i is incremented by 1;
if destination address $j < i$, then BQ_i is incremented by 1.

The structure of the AC field is shown in Fig. 3. Let t_{fn} be the start of the nth frame. The AC field is from t_{fn} to t_{bn}. Station S_i gains access to line A in the following manner. Let S_i be permitted access for p_{max} packets each cycle. At t_{fn}, the start of the nth frame, the read tap reads the START bit of the AC field. The start of cycle is indicated by START = 1. Because of gate delays in the decision circuitry and propagation delays in the tap cables, the outcome of the read operation is only known at t_{sn}. This additional time of duration τ_{dec}, shown in Fig. 3, is of the order of a few bit times for a 100 Mb/s line and nanosecond logic. Next, the station may simultaneously read the BUSY bit via the read tap and write BUSY = 1 via the directional coupler. Again, the outcome of the read operation is only known at t_{bn} after a delay of τ_{dec}. Nonetheless, the write operation does not alter the BUSY bit if it is already set to 1. The nature of the signaling to achieve this is explained in Section 5.2. At t_{bn}, if BUSY = 1, the station defers until the BUSY bit of the next frame. If BUSY = 0, the station accesses the line for the remaining frame duration.

Station S_i is said to be in one of four states:
IDLE —if it has no packets to transmit, i.e., $AQ_i = 0$.
WAIT—if it is waiting for the start of cycle.

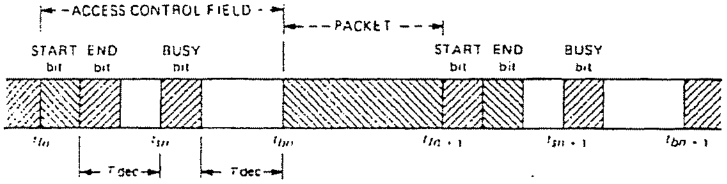


Fig. 3—The frame structure of Fasnet. Each frame consists of (*i*) an access control field containing START, END, and BUSY bits; and (*ii*) the packet as provided by the logical link sublayer.

DEFER —if it is deferring to busy users who are upstream on the line.

ACCESS—if it is accessing the line.

The station makes transitions (denoted as →) between states as follows (Fig. 4): While $AQ_i = 0$, S_i is in IDLE. Upon arrival of a packet for line A, $AQ_i > 0$ and $S_i \rightarrow$ WAIT. The station reads the START bit of every frame. When START = 1 $S_i \rightarrow$ DEFER, and the station simultaneously reads and writes the BUSY bit as described above for every frame. When BUSY = 0 $S_i \rightarrow$ ACCESS. Now it accesses the line for p_{max} frames and also writes BUSY = 1 for each. Then $S_i \rightarrow$ WAIT. The station may cease to access the line earlier if $AQ_i = 0$, whereby S_i → IDLE.

Station S_1 initiates cycles by START = 1 in the first frame of each cycle. There is an additional bit, END, in each frame to indicate the end of cycles. This bit can be conveniently located in the blank portion of the frame after the START or BUSY bits. When station S_N reads BUSY = 0 on line A (indicating that all active stations have accessed the line), it sets END = 1 in the next frame on line B. On receipt of this frame on line B, S_1 then initiates a new cycle on line A. Thus, in the worst case, line A will be silent once every cycle for a time equal to twice the end-end propagation delay, plus as each end station has to wait until the nex or END bits.

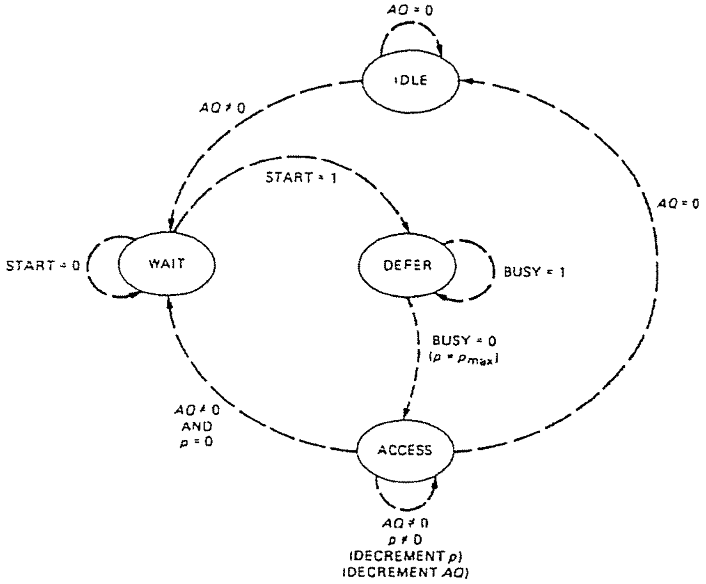


Fig. 4—State transition diagram describing the operation of a Fasnet station.

The operation on line B is identical, wi reversed. Thus, the two lines cycle indepen access being passed between the stations i physical locations on each line.

In the protocol described above, the outc on the START bit needs to be known befoı so that the first frame of a new cycle does n START and BUSY bits be adjacent to eacl learn that START = 1 after the BUSY bit will not be used. However, for large cycle lengths, the reduction of one decision inteı be greater than the addition of the extra idl

A further alternative is to have the first f only an access field. However, unequal fr chronization for a very small increase in eff

3.3 Error recovery

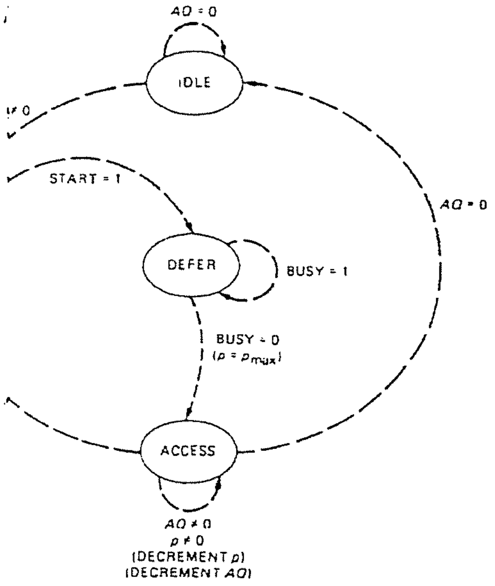
The protocol is controlled by the STAR An error in a BUSY field will have no lasti packet being overwritten if the busy bit is Alternatively, an empty slot will go unused from 0 to 1. Of more significance is an erroı fields. If a START field is set to 1 in error, ı and an END would be simultaneously pres

It will be shown that generation of addit will not propagate and have little effect on We will assume that end stations do not ge that are closer together than the round-t normal operation this cannot occur. A fal either in the active portion of a cycle (incl or in the empty slots occurring at the end a new cycle will start before, or as, the old additional START = 1 will not generate a line (because no transition from busy slot the condition will not propagate. If the false empty slots, other than the first, the new c (actually increasing utilization temporaril results:

(*i*) The busy part of the additional cyc slot before the next normally occurring ST end station will detect an end condition. H eferring to busy users who are upstream on the

cessing the line.
transitions (denoted as →) between states as
$AQ_i = 0$, S_i is in IDLE. Upon arrival of a packet
d $S_i \rightarrow$ WAIT. The station reads the START bit
n START = 1 $S_i \rightarrow$ DEFER, and the station
and writes the BUSY bit as described above for
USY = 0 $S_i \rightarrow$ ACCESS. Now it accesses the line
so writes BUSY = 1 for each. Then $S_i \rightarrow$ WAIT.
e to access the line earlier if $AQ_i = 0$, whereby S_i

cycles by START = 1 in the first frame of each
litional bit, END, in each frame to indicate the
can be conveniently located in the blank portion
START or BUSY bits. When station S_N reads
indicating that all active stations have accessed
= 1 in the next frame on line B. On receipt of
S_1 then initiates a new cycle on line A. Thus, in
will be silent once every cycle for a time equal to twice the end-end propagation delay, plus twice the frame duration, as each end station has to wait until the next frame to set the START or END bits.



n diagram describing the operation of a Fasnet station.

The operation on line B is identical, with the roles of S_1 and S_N reversed. Thus, the two lines cycle independently of each other with access being passed between the stations in the same order as their physical locations on each line.

In the protocol described above, the outcome of the read operation on the START bit needs to be known before the BUSY bit is written so that the first frame of a new cycle does not remain idle. Should the START and BUSY bits be adjacent to each other, a station will only learn that START = 1 after the BUSY bit has passed and the frame will not be used. However, for large cycle lengths and short packet lengths, the reduction of one decision interval, τ_{dec}, per frame would be greater than the addition of the extra idle frame.

A further alternative is to have the first frame of each cycle contain only an access field. However, unequal frame sizes complicate synchronization for a very small increase in efficiency.

3.3 Error recovery

The protocol is controlled by the START, BUSY, and END fields. An error in a BUSY field will have no lasting effect; it will result in a packet being overwritten if the busy bit is changed from a 1 to a 0. Alternatively, an empty slot will go unused if the busy bit is changed from 0 to 1. Of more significance is an error in the START and END fields. If a START field is set to 1 in error, two STARTs or a START and an END would be simultaneously present on the loop.

It will be shown that generation of additional STARTs and ENDs will not propagate and have little effect on the operation of the link. We will assume that end stations do not generate STARTs or ENDs that are closer together than the round-trip delay time, τ_r; under normal operation this cannot occur. A false START = 1 will occur either in the active portion of a cycle (including the first empty slot) or in the empty slots occurring at the end of the cycle. If the former, a new cycle will start before, or as, the old one is finishing. Since the additional START = 1 will not generate an END = 1 on the return line (because no transition from busy slot to empty slot is detected), the condition will not propagate. If the false START = 1 occurs in the empty slots, other than the first, the new cycle will start prematurely (actually increasing utilization temporarily). One of two conditions results:

(*i*) The busy part of the additional cycle terminates at least one slot before the next normally occurring START = 1, in which case the end station will detect an end condition. However, because the period since the last END = 1 is less than τ_r a new END = 1 will not be generated.

(ii) There is no empty slot before the next normally occurring START = 1. As a result, an additional end condition is not detected by the end station.

Thus, a START = 1 resulting from a fault condition will not produce additional END = 1 bits on the return line. On the other hand, END = 1 faults, unless they are closer together than τ_r, will produce additional START = 1 slots which as just described, have a transient effect on the operation of the link.

Consider the condition where a START or END bit is changed from a 1 to a 0. A new cycle would fail to initiate. After a time-out greater than the longest permitted cycle time, the head station will issue a START = 1, and the link will continue to operate normally. Should a head or end station fail, the station next to the head or end station would assume the functions on detecting loss of timing or after timing out on the arrival of START = 1 or END = 1.

3.4 Fault diagnosis

The independent lines of the Fasnet link provide the opportunity to localize and mitigate some types of faults. Consider first that a line is severed because of some catastrophic event or something trivial like a cable connector failing. The result will usually be either a short or open circuit leading to a gross impedance mismatch. The fault will most likely terminate all effective communication on the upstream side of the fault because of reflections from the mismatch traveling back into station interface units via the read tap. The downstream segment will be affected very little because the directional couplers will propagate little energy in the direction of the mismatch. Thus, a diagnostic program in the end station can determine between which stations the mismatch lies. This is done by having the end station send a query to each station via the intact line and determining which stations respond to the query.

A difficult type of station fault is to have a station continuously write garbage on a line. Diagnostic programs in the end stations, again by querying each station, can determine the faulty station and remove it from service. The head station on the line with the fault, after being informed of the fault by the end station, via the other line, queries each station in turn. If the station fault is confined only to the write circuit, the faulty station will respond. The next station on the downstream side will not respond, since it will not be able to read the query sent by the head station because of the interference from the faulty station. If both read and write circuits in the faulty station are affected, the last correctly responding station will be the station on the upstream side of the faulty station. Thus, the fault i stations. Both stations may then be disconnec signal sent via the functioning line. The fa uniquely determined by returning one of the fault condition resumes, the returned statio nected; otherwise the other station is faulty.

IV. PERFORMANCE

4.1 Sample operation

Typical operation of Fasnet for lines of 2.5- Mb/s bandwidth, and 200-bit frame length is the time-space relation of the frames on eacl represents time divided into slots A_1, A_2, A_3, B_3, ··· for line B. The vertical axis represe of the active stations S_1, S_2, S_3, S_4, and S additionally, as end stations. The electrical Station S_1 initiates the cycle in frame A_1, an S_2 to S_3 to S_4. When the end station, S_5, sens it sets END = 1 in frame B_9. Receipt of thi

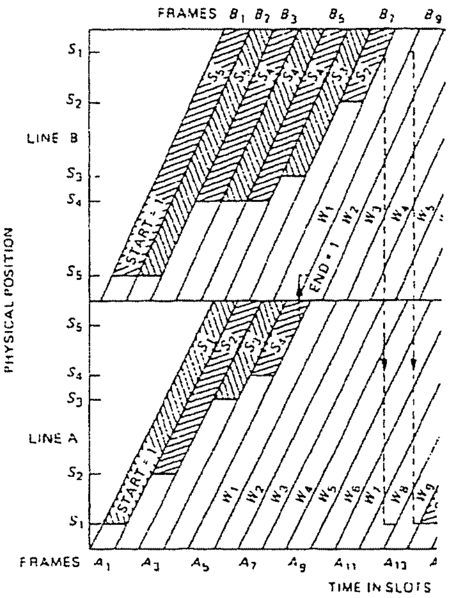

FRAMES B_1 B_2 B_3 B_5 B_7 B_9
S_1
S_2
LINE B
S_3
S_4
S_5
S_5 S_5 S_4 S_4 S_4 S_3 S_2
START = 1
END = 1
W_1 W_2 W_3 W_4 W_5
PHYSICAL POSITION
S_5
S_4
S_3
LINE A
S_2
S_1
S_1 S_2 S_3 S_4
START = 1
W_1 W_2 W_3 W_4 W_5 W_6 W_7 W_8 W_9
FRAMES A_1 A_3 A_5 A_7 A_9 A_{11} A_{13} A
TIME IN SLOTS

Fig. 5—A graph of activity on a Fasnet link (lines A dotted lines indicate the flow of information from one

1 is less than τ_r a new END = 1 will not be

ıpty slot before the next normally occurring ılt, an additional end condition is not detected

esulting from a fault condition will not produce ts on the return line. On the other hand, END ey are closer together than τ_r, will produce slots which as just described, have a transient of the link.

on where a START or END bit is changed from would fail to initiate. After a time-out greater itted cycle time, the head station will issue a nk will continue to operate normally. Should a il, the station next to the head or end station tions on detecting loss of timing or after timing TART = 1 or END = 1.

es of the Fasnet link provide the opportunity to ome types of faults. Consider first that a line is e catastrophic event or something trivial like a . The result will usually be either a short or a gross impedance mismatch. The fault will ıall effective communication on the upstream ıse of reflections from the mismatch traveling face units via the read tap. The downstream ed very little because the directional couplers ergy in the direction of the mismatch. Thus, a the end station can determine between which ies. This is done by having the end station send ın via the intact line and determining which query.

tation fault is to have a station continuously Diagnostic programs in the end stations, again n, can determine the faulty station and remove ıd station on the line with the fault, after being y the end station, via the other line, queries the station fault is confined only to the write n will respond. The next station on the downond, since it will not be able to read the query n because of the interference from the faulty write circuits in the faulty station are affected, ding station will be the station on the upstream side of the faulty station. Thus, the fault is isolated to one of two stations. Both stations may then be disconnected by means of a control signal sent via the functioning line. The faulty station may then be uniquely determined by returning one of the stations to service. If the fault condition resumes, the returned station is faulty and is disconnected; otherwise the other station is faulty.

IV. PERFORMANCE

4.1 Sample operation

Typical operation of Fasnet for lines of 2.5-km individual length, 100 Mb/s bandwidth, and 200-bit frame length is shown in Fig. 5. It shows the time-space relation of the frames on each line. The horizontal axis represents time divided into slots $A_1, A_2, A_3, \cdots$ for line A and $B_1, B_2, B_3, \cdots$ for line B. The vertical axis represents the physical locations of the active stations S_1, S_2, S_3, S_4, and S_5 with S_1 and S_5 serving, additionally, as end stations. The electrical line length is five frames. Station S_1 initiates the cycle in frame A_1, and access passes from S_1 to S_2 to S_3 to S_4. When the end station, S_5, senses BUSY = 0 in frame A_5, it sets END = 1 in frame B_9. Receipt of this frame by S_1 causes it to

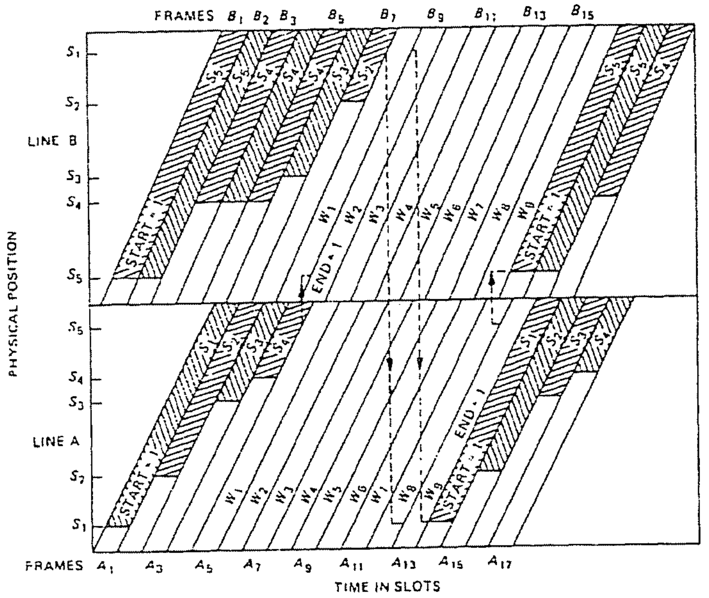


Fig. 5—A graph of activity on a Fasnet link (lines A and B) as a function of time. The dotted lines indicate the flow of information from one line to another.

initiate a new cycle in A_{14}. Similarly, a cycle on line B starts at B_1. Assume that S_5 and S_4 are permitted access for up to two and three packets, respectively. Station S_1 senses BUSY = 0 in frame B_8 and sets END = 1 in A_{13}. Receipt of this frame by S_5 causes it to start a new cycle in B_{17}.

4.2 Utilization

As there are no collisions, no capacity is lost through collision resolution. However, the utilization is not 100 percent as each line is idle at the end of each cycle. The idle period is nine frames, W_1, W_2, $\cdots$ W_9, on each line in Fig. 4. In the worst case, this is equal to twice the end-end propagation delay, plus twice the slot time (one slot time on average) as each end station has to wait until the start of the next slot to set the START or END bits. If

ν = speed of propagation on the line (m/s)
W = line capacity (b/s)
L = line length (m)
F = frame size (bits)
M = number of busy stations with downstream traffic,

then if each station is allowed access for only a single packet per cycle,

$$\text{cycle length} = \tau_c = M^*(F/W) + 2^*(L/\nu) + (F/W)\,,$$

$$\text{duration of busy frames} = \tau_b = M^*(F/W)\,, \tag{1}$$

and

$$\text{utilization} = \eta = \frac{M^*(F/W)}{M^*(F/W) + 2^*(L/\nu) + (F/W)}. \tag{2}$$

The effective utilization is lower since a fraction of each frame is devoted to access control. However, for large F, it results in only a small reduction in utilization. If $\nu = 2.5 \times 10^8$ m/s, $W = 100 \times 10^6$ b/s, $L = 2.5 \times 10^3$ m,

$$\eta = \frac{M^*F}{(M+1)^*F + 2000}.$$

For $M = 100$,

$F = 50$	$\eta = 71$ percent
$F = 100$	$\eta = 83$ percent
$F = 200$	$\eta = 90$ percent
$F = 500$	$\eta = 95$ percent
$F = 1000$	$\eta = 97$ percent.

For the same values of ν, W, and L, with ĩ Ethernet slot time $T = 50 \times 10^{-6}$ s,[3] we can of Fasnet and Ethernet as the number of s in Table II.

Unlike Ethernet, Fasnet has the desirab increases, the utilization also increases. T reflect the fact that in practice the length consequently, the fixed frames of Fasnet fı tially filled. The effect on η depends on the and to some extent is determined by the sy in a system designed for large amounts of v equal to the size of a voice packet.

V. IMPLEMENTATION CONSIDERATIONS

The design criteria previously stressed in implementation in important ways. In part operate at high speeds and the unidirecti affect the design of the synchronization sy synchronization and the use of directional of the line code that is used.

5.1 Synchronization

Bus systems in which signals travel in I require the receiving stations to adapt to the sending station because the amplitude, the received signal vary depending upon tl ting station on the line. Synchronization ca when the signaling rate is low relative to t mission medium. At higher signaling rates

Table II—Fasnet ve
Ethernet as a functi
number of busy sta

	Fasnet	Et
M	(in perce	
5	50	
10	67	
50	91	
100	95	

* Note that since the n permissible packet length bits, η is calculated as 0.1 · 5000-bit packets. Other c: tocols that do not require detection perform better.

e in A_{14}. Similarly, a cycle on line B starts at B_1. d S_4 are permitted access for up to two and three ly. Station S_1 senses BUSY = 0 in frame B_8 and 13. Receipt of this frame by S_5 causes it to start a

collisions, no capacity is lost through collision r, the utilization is not 100 percent as each line is ach cycle. The idle period is nine frames, W_1, W_2, e in Fig. 4. In the worst case, this is equal to twice ation delay, plus twice the slot time (one slot time i end station has to wait until the start of the next RT or END bits. If

of propagation on the line (m/s)

pacity (b/s)

ngth (m)

size (bits)

r of busy stations with downstream traffic,

is allowed access for only a single packet per cycle,

$$\text{length} = \tau_c = M^*(F/W) + 2^*(L/\nu) + (F/W)\,,$$

$$\text{frames} = \tau_b = M^*(F/W)\,, \tag{1}$$

$$\text{ization} = \eta = \frac{M^*(F/W)}{M^*(F/W) + 2^*(L/\nu) + (F/W)}\,. \tag{2}$$

ization is lower since a fraction of each frame is ontrol. However, for large F, it results in only a tilization. If $\nu = 2.5 \times 10^8$ m/s, $W = 100 \times 10^6$ b/s,

$$\eta = \frac{M^*F}{(M+1)^*F + 2000}\,.$$

$F = 50$ $\eta = 71$ percent

$F = 100$ $\eta = 83$ percent

$F = 200$ $\eta = 90$ percent

$F = 500$ $\eta = 95$ percent

$F = 1000$ $\eta = 97$ percent.

For the same values of ν, W, and L, with $F = 500$[1,2] and assuming an Ethernet slot time $T = 50 \times 10^{-6}$ s,[3] we can compare the performance of Fasnet and Ethernet as the number of stations is varied as shown in Table II.

Unlike Ethernet, Fasnet has the desirable feature that as the load increases, the utilization also increases. The above figures do not reflect the fact that in practice the length of packets is variable and, consequently, the fixed frames of Fasnet frequently will be only partially filled. The effect on η depends on the distribution of packet size, and to some extent is determined by the system design. For example, in a system designed for large amounts of voice traffic, F could be set equal to the size of a voice packet.

V. IMPLEMENTATION CONSIDERATIONS

The design criteria previously stressed in the introduction affect the implementation in important ways. In particular, the requirement to operate at high speeds and the unidirectional operation of the bus affect the design of the synchronization system; in turn, the type of synchronization and the use of directional couplers impact the choice of the line code that is used.

5.1 Synchronization

Bus systems in which signals travel in both directions on the line require the receiving stations to adapt to the signals transmitted by the sending station because the amplitude, dispersion, and phasing of the received signal vary depending upon the position of the transmitting station on the line. Synchronization can be achieved very quickly when the signaling rate is low relative to the bandwidth of the transmission medium. At higher signaling rates, synchronization needs to

Table II—Fasnet versus Ethernet as a function of number of busy stations

M	Fasnet (in percent)	Ethernet* (in percent)
5	50	4.1
10	67	3.9
50	91	3.7
100	95	3.7

* Note that since the minimum permissible packet length is 5000 bits, η is calculated as 0.1 of η with 5000-bit packets. Other CSMA protocols that do not require collision detection perform better.

be more accurate to achieve good error performance. Ethernet specifies a synchronization preamble of 64 bits and for higher transmission rates an even longer sequence may be required. Thus, for short messages efficiency would be significantly reduced. Using a unidirectional bus, each station can be synchronized to a common clock issued from the head station. Thus, if all stations add signals to the cable in phase with the transmitting clock, stations will receive the signals in correct phase. Similarly, fixed gain and frequency compensation can be employed. The problem of reliability can be overcome by giving each station the ability to supply clock. The clock drive would be inhibited by detection of, and locking to, an incoming clock.

Initial tests have shown that a simple, cost-effective method of synchronization is to synchronize to a continuously injected pilot tone placed at the high end of the signaling band. The synchronizing function then assumes a negligible fraction of the transmission capacity.

In addition to bit synchronization, frame synchronization is also required. This is achieved by sending periodically a synchronizing bit pattern. Design is simplified if this is sent after an integral number of frames, say 64 or 128. With tight bit and frame synchronization, successive frames may be butted together without a gap.

5.2 Signaling

Because synchronization is achieved independently of the data signal, line codes with fewer transitions may be considered. It is particularly convenient if a code is chosen that couples no energy to the line when one of the logic states is continuously transmitted (assume logic 0). Each station at the end of transmission then simply returns to logic 0, and there is no need to "disconnect" the transmitter from the line. The two line codes we are investigating are a bipolar three-level code, Fig. 6a, and a nonreturn to zero (NRZ) two-level code, Fig. 6b. The two-level signal has a greater noise margin; however, one has to contend with the dc signal component.[14]

The Fasnet protocol does not permit subsequent stations to modify a signal already transmitted by an upstream source. In principle, this could be done. For example, a signal from one station could be deleted from the line by a second station writing the complement on the line. In practice, signal levels would have to be matched very accurately for such a scheme to work.

There is one condition in which more than one station may add energy to the same bit in a frame—the BUSY bit of the AC field. As a result, the amplitude of this bit may far exceed the amplitude of the remaining signal. This may lead to errors in adjacent bits of the AC field. To prevent this, guard bands on either side of the BUSY bit should be used. Notice from Fig. 3 that the a
so that the guard bands fall in the intervals ı
have a comparable duration.

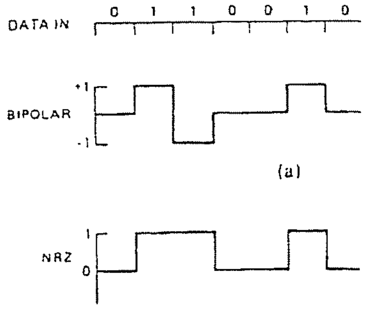


Fig. 6—(a) Bipolar three-level line code. (b) NRZ

VI. IMPROVING UTILIZATION

As can be seen from (1), efficiency incre
increases and (*ii*) as the idle period at the end
gap) decreases. At the expense of some incre
niques may be devised to improve utilization b
or reducing intercycle gap.

6.1 Control of cycle length

Since START = 1 may be read by all static
cycle, τ_c, may be determined by any station.
each station may transmit up to p_{max} packe
controlling p_{max}, stations may influence the va
of τ_c by manipulation of p_{max} is obviously lim
assume that p is fixed at 1 and that we have
packets at a rate $< 1/\tau_c$. Increasing p will not
since packets will be transmitted before a
other hand, increasing p for heavily loaded
increase of τ_c, provided τ_c is less than the acc

6.2 Reducing inter-cycle gap

Three methods are described for reducin
hence increasing the line utilization. In the fi
END = 1 bit seize empty slots on the other li
use the END field as a request field; in th

chieve good error performance. Ethernet specifies amble of 64 bits and for higher transmission rates nce may be required. Thus, for short messages ignificantly reduced. Using a unidirectional bus, ynchronized to a common clock issued from the all stations add signals to the cable in phase with k, stations will receive the signals in correct phase. and frequency compensation can be employed. pility can be overcome by giving each station the . The clock drive would be inhibited by detection incoming clock.

shown that a simple, cost-effective method of synchronize to a continuously injected pilot tone end of the signaling band. The synchronizing es a negligible fraction of the transmission ca-

synchronization, frame synchronization is also eved by sending periodically a synchronizing bit plified if this is sent after an integral number of 8. With tight bit and frame synchronization, y be butted together without a gap.

zation is achieved independently of the data h fewer transitions may be considered. It is t if a code is chosen that couples no energy to f the logic states is continuously transmitted station at the end of transmission then simply there is no need to "disconnect" the transmitter o line codes we are investigating are a bipolar a, and a nonreturn to zero (NRZ) two-level code, signal has a greater noise margin; however, one e dc signal component.[14]

does not permit subsequent stations to modify nitted by an upstream source. In principle, this mple, a signal from one station could be deleted nd station writing the complement on the line. s would have to be matched very accurately for

ion in which more than one station may add in a frame—the BUSY bit of the AC field. As a f this bit may far exceed the amplitude of the may lead to errors in adjacent bits of the AC guard bands on either side of the BUSY bit should be used. Notice from Fig. 3 that the access field is configured so that the guard bands fall in the intervals τ_{dec} and, in practice, will have a comparable duration.

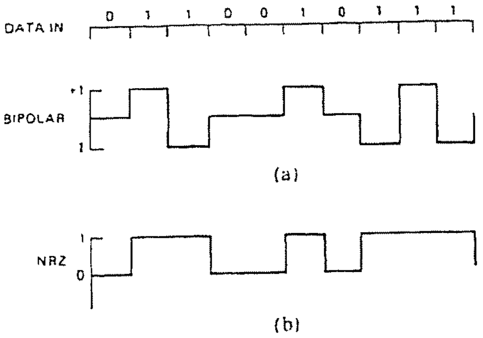


Fig. 6—(a) Bipolar three-level line code. (b) NRZ two-level line code.

VI. IMPROVING UTILIZATION

As can be seen from (1), efficiency increases (*i*) as cycle length increases and (*ii*) as the idle period at the end of each cycle (intercycle gap) decreases. At the expense of some increase in complexity, techniques may be devised to improve utilization by increasing cycle length or reducing intercycle gap.

6.1 Control of cycle length

Since START = 1 may be read by all stations, the length of the last cycle, τ_c, may be determined by any station. As previously described, each station may transmit up to p_{max} packets per access. Thus, by controlling p_{max}, stations may influence the value of τ_c. Station control of τ_c by manipulation of p_{max} is obviously limited. For example, let us assume that p is fixed at 1 and that we have stations each generating packets at a rate $< 1/\tau_c$. Increasing p will not change the cycle length since packets will be transmitted before a queue can form. On the other hand, increasing p for heavily loaded stations will lead to an increase of τ_c, provided τ_c is less than the accepted maximum.

6.2 Reducing inter-cycle gap

Three methods are described for reducing the intercycle gap and hence increasing the line utilization. In the first, stations detecting the END = 1 bit seize empty slots on the other line; in the second, stations use the END field as a request field; in the third, the end station

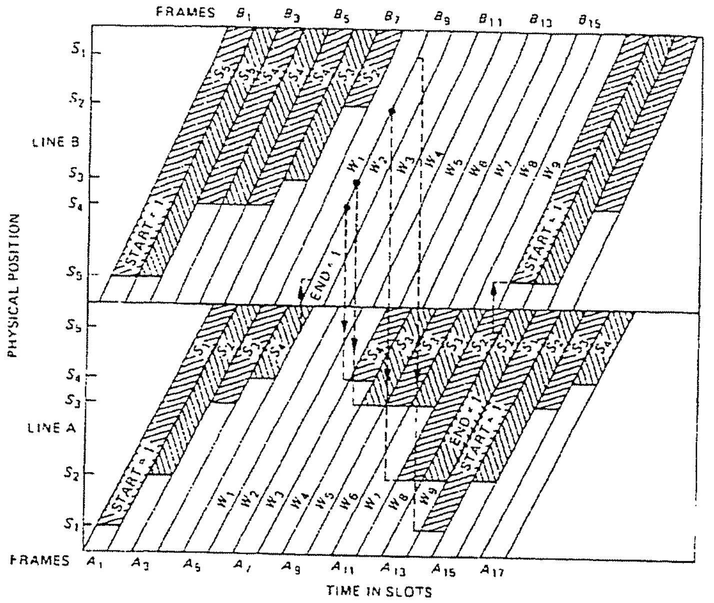

FRAMES
B_1 B_3 B_5 B_7 B_9 B_{11} B_{13} B_{15}
S_1
S_2
LINE B
S_3
S_4
S_5
PHYSICAL POSITION
START = 1
END = 1
W_1 W_2 W_3 W_4 W_5 W_6 W_7 W_8 W_9
S_5
S_4
S_3
LINE A
S_2
S_1
FRAMES
A_1 A_3 A_5 A_7 A_9 A_{11} A_{13} A_{15} A_{17}
TIME IN SLOTS

Fig. 7—A graph of activity on a Fasnet link as a function of time. Empty slots encountered on line A by stations that have read END = 1 on line B are utilized. The IDLE period is reduced to three slots. Corresponding use of IDLE slots on line B would also occur but is not shown.

attempts to estimate the end of a cycle, setting END = 1 before BUSY = 0 is received.

Considering the first method, any station S_i in the WAIT state that observes END = 1 may attempt to seize any empty slots on the opposite line.[12] The number of empty slots seized depends on the time the END = 1 frame takes to propagate to the next active station, which then seizes empty slots, thus preempting active stations downstream.* The intercycle gap now depends on the propagation time from the last active station to the end station and back. (The relative timing of the frame starts in the two lines will also affect the gap size). As shown in the example of Fig. 7, the intercycle gap has been reduced from nine slots to three, shown for line A only.

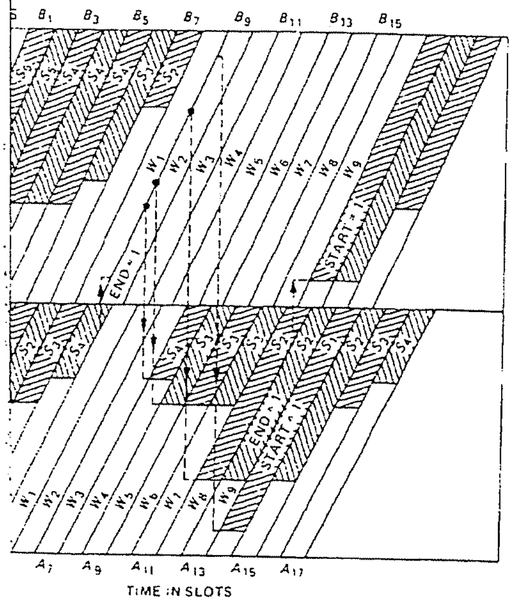
In the second method, stations in DEFER and ACCESS states write REQUEST (REQ) = 1 on the return line (the END field is now replaced by an REQ field). After all stations
head station will read REQ = 0 and initiate a c
least one REQ = 1 occurs in a cycle for whi
active, a station in changing from WAIT to
writes at least one REQ = 1. Approximately
intercycle gap is now equal to twice the prop
head station to the last active station, plus half
this procedure is more distributed in that the
recognizing the end-of-cycle END condition ar
the return line is now bypassed. Each station r
alent operation.

A further refinement is to observe that th
can also be distributed.* Each station with traf
wait for the head station to issue START = :
REQ = 0 on the return line, it can switch fro
ACCESS mode setting the p register to the allo
that may be transmitted per cycle. The int
eliminated under heavy traffic conditions anc
users as in the first method. For two continuou
intercycle gap is twice the delay time betwee
slot time on average. Notice that even thou
issued, or is necessary, the loop cycles. Howeve
packets from a source during one cycle will usu
The station protocol is summarized in Fig. 8
diagram. As seen in comparison with Fig. 4
complex; however, the algorithm no longer
head or end-station functions. The issue of di
centralized control is particularly important
dating different types of traffic and is discusse

In the third method, each active statio
indicates its desire to transmit by setting an a
1 (REQ = 1) in the access field of the line on
as in Ref. 12. The end station estimates the cy
END = 1 timed to arrive at the head station a
in the cycle is leaving the head station. If the
end station will read REQ = 1 in the last fram
that the estimate of the length of the la
Therefore, the estimate of the length of t
increased. If the estimate is too high, there w
the arrival of the next START = 1, and the
the next cycle would be decreased. If the est

* A station may only transmit a single frame at a time because of the possibility of preemption. This will interfere with construction of "superpackets." Superpackets are used to increase efficiency by reducing the effect of packet overhead. The overhead is attached only to the first packet of the superpacket.[12]

* Suggested by Z. L. Budrikis in connection with tl
Fasnet.



ivity on a Fasnet link as a function of time. Empty slots stations that have read END = 1 on line B are utilized. The three slots. Corresponding use of IDLE slots on line B would n.

the end of a cycle, setting END = 1 before BUSY

st method, any station S_i in the WAIT state that may attempt to seize any empty slots on the umber of empty slots seized depends on the time takes to propagate to the next active station, pty slots, thus preempting active stations down-cle gap now depends on the propagation time tation to the end station and back. (The relative arts in the two lines will also affect the gap size). ple of Fig. 7, the intercycle gap has been reduced ee, shown for line A only.

od, stations in DEFER and ACCESS states write 1 on the return line (the END field is now replaced by an REQ field). After all stations have been served, the head station will read REQ = 0 and initiate a cycle. To ensure that at least one REQ = 1 occurs in a cycle for which only one station is active, a station in changing from WAIT to DEFER or ACCESS writes at least one REQ = 1. Approximately speaking, the average intercycle gap is now equal to twice the propagation time from the head station to the last active station, plus half a slot time. Notice that this procedure is more distributed in that the end-station function of recognizing the end-of-cycle END condition and writing END = 1 on the return line is now bypassed. Each station now performs an equivalent operation.

A further refinement is to observe that the head-station function can also be distributed.* Each station with traffic to transmit need not wait for the head station to issue START = 1. Rather, after reading REQ = 0 on the return line, it can switch from WAIT to DEFER or ACCESS mode setting the *p* register to the allowed number of packets that may be transmitted per cycle. The intercycle gap is virtually eliminated under heavy traffic conditions and for a large number of users as in the first method. For two continuously queued stations, the intercycle gap is twice the delay time between the stations, plus one slot time on average. Notice that even though no START is being issued, or is necessary, the loop cycles. However, as in the first method, packets from a source during one cycle will usually not be consecutive. The station protocol is summarized in Fig. 8 by means of the state diagram. As seen in comparison with Fig. 4, the protocol is more complex; however, the algorithm no longer requires the centralized head or end-station functions. The issue of distributed control versus centralized control is particularly important in efficiently accommodating different types of traffic and is discussed further in Section VII.

In the third method, each active station in the DEFER state indicates its desire to transmit by setting an additional request field to 1 (REQ = 1) in the access field of the line on which it wishes to write, as in Ref. 12. The end station estimates the cycle length and transmits END = 1 timed to arrive at the head station as the last slot to be used in the cycle is leaving the head station. If the estimate was too low, the end station will read REQ = 1 in the last frame in the cycle indicating that the estimate of the length of the last cycle was too short. Therefore, the estimate of the length of the next cycle would be increased. If the estimate is too high, there will be empty slots prior to the arrival of the next START = 1, and the estimate of the length of the next cycle would be decreased. If the estimate is correct, then the

nsmit a single frame at a time because of the possibility of fere with construction of "superpackets." Superpackets are by reducing the effect of packet overhead. The overhead is acket of the superpacket.[12]

[*] Suggested by Z. L. Budrikis in connection with the earlier single-line version of Fasnet.

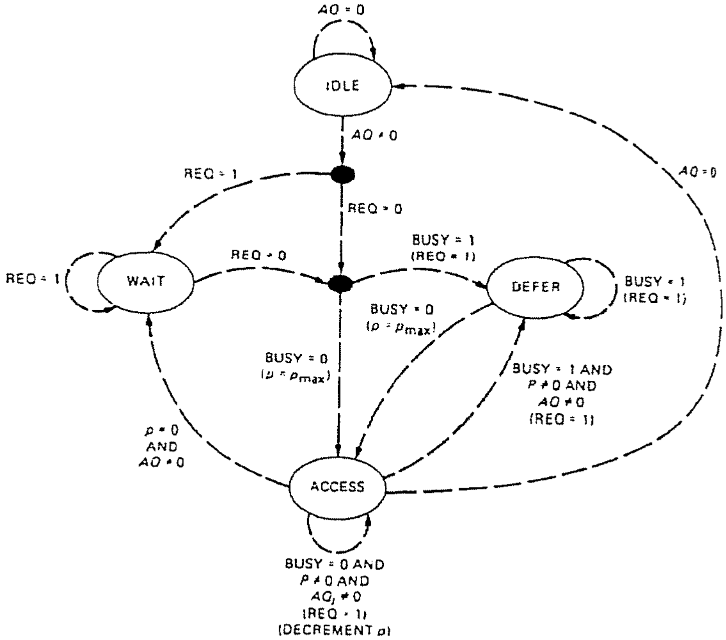


Fig. 8—State transition diagram of a distributed version of the Fasnet protocol in which unique functions in the head and end stations are not employed. Statements in parentheses are actions executed on making the transition. The filled circles denote intermediate states.

last frame before the next START = 1 will have REQ = 0 and the estimate of cycle length would remain unchanged.

Of the three methods for reducing intercycle gap, the former two are distributed while the latter requires additional intelligence in the end station.

VII. TRAFFIC CONTROL

Different types of traffic have very different transmission requirements. Voice traffic is an interesting example of where an initial restriction of access (blocking) is preferable to losing voice packets. Real-time traffic such as voice and video requires a guaranteed maximum delay if information is not to be lost. In contrast, some types of computer traffic are rather tolerant of delay. It is not our aim to develop here a comprehensive algorithm for handling different traffic types; it would depend closely on the particular environment and the mix of traffic. However, we would like to indicate the mechanisms that Fasnet can support to control integration of traffic.

We can identify four different types of cont

(*i*) Selection of traffic—The ability to inc of traffic.

(*ii*) Request for access—The ability to cc is required.

(*iii*) Blocking of traffic—The ability to pre class from gaining access to the system.

(*iv*) Continuation of service—The ability transmission. We will take these four control consider how they may be implemented in Fa

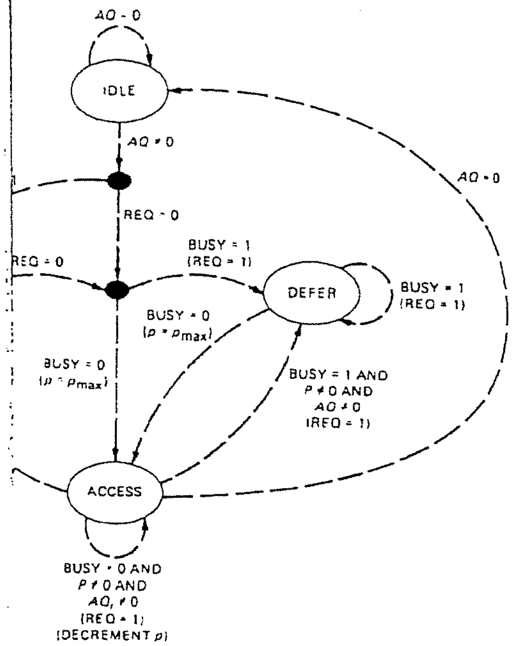
(*i*) Selection—We can borrow from the s al.[15] and Ulug et al.[6] START, instead of be expanded to a multibit word. A START code each class of traffic. A station would then t only if the START corresponded to the class transmit. A class of traffic could denote a priority. The term subcycle will be used to START of one class to the next occurring reserved for the period between two START

(*ii*) Request—The amount of request in knowing exactly which station wants to s extreme to knowing just that a station somev type of traffic in the other extreme. A compro a request word adjacent to the END field in line. Each bit in the request word would den information would enable the head station more stations required service of a particula the extent of the demand. Information about most likely be used to adapt the control st there was a change in the balance between t

(*iii*) Blocking—The channel capacity all controlled by the time allocated to that clas al.[15] By issuing a different START after a gi of the original class type wishing to transmi Blocking is typically used to handle overflow generates information periodically, such as and synchronous data traffic. In this insta speak of connections between source and guaranteed access once a connection has be

To discuss the allocation of connections requires that the previous definitions of W state be generalized:

WAIT—A station is waiting for permissio



iagram of a distributed version of the Fasnet protocol in
ie head and end stations are not employed. Statements in
cuted on making the transition. The filled circles denote

next START = 1 will have REQ = 0 and the
n would remain unchanged.
ls for reducing intercycle gap, the former two
ne latter requires additional intelligence in the

affic have very different transmission require-
an interesting example of where an initial
locking) is preferable to losing voice packets.
s voice and video requires a guaranteed maxi-
on is not to be lost. In contrast, some types of
ither tolerant of delay. It is not our aim to
ensive algorithm for handling different traffic
closely on the particular environment and the
we would like to indicate the mechanisms that
ontrol integration of traffic.

We can identify four different types of control mechanisms.

(*i*) Selection of traffic—The ability to individually control a class of traffic.

(*ii*) Request for access—The ability to communicate that service is required.

(*iii*) Blocking of traffic—The ability to prevent traffic of a specific class from gaining access to the system.

(*iv*) Continuation of service—The ability to delay traffic for later transmission. We will take these four control mechanisms in turn and consider how they may be implemented in Fasnet.

(*i*) Selection—We can borrow from the strategy used by Frata et al.[15] and Ulug et al.[6] START, instead of being a single bit, can be expanded to a multibit word. A START code can then be allocated to each class of traffic. A station would then be permitted to transmit only if the START corresponded to the class of traffic it is waiting to transmit. A class of traffic could denote a traffic type, as well as a priority. The term subcycle will be used to denote the period from a START of one class to the next occurring START. Cycle will be reserved for the period between two STARTs of the same class.

(*ii*) Request—The amount of request information can vary from knowing exactly which station wants to send what traffic in one extreme to knowing just that a station somewhere wants to send some type of traffic in the other extreme. A compromise would be to provide a request word adjacent to the END field in each packet on the return line. Each bit in the request word would denote a class of traffic. This information would enable the head station to determine that one or more stations required service of a particular type without indicating the extent of the demand. Information about demand for service would most likely be used to adapt the control strategy in the case where there was a change in the balance between traffic types.

(*iii*) Blocking—The channel capacity allocated to a class can be controlled by the time allocated to that class as suggested in Frata et al.[15] By issuing a different START after a given period, further traffic of the original class type wishing to transmit would be denied access. Blocking is typically used to handle overflow of the type of traffic that generates information periodically, such as real-time voice and video, and synchronous data traffic. In this instance, it is appropriate to speak of connections between source and destination which implies guaranteed access once a connection has been allocated.

To discuss the allocation of connections in a blocking traffic class requires that the previous definitions of WAIT state and ACCESS state be generalized:

WAIT—A station is waiting for permission to seek access to the line.

ACCESS—A station has a connection.

Stations that already have access take the first free slot available to them after the appropriate START. Allocation of freed up slots on a reasonably equitable basis would proceed as follows. Stations would be aware of say n slots becoming free from the position of the END bit on the return line. Note, no END is issued if the class is full. Stations in the DEFER state would be permitted to compete for the n empty slots at the end of the subcycle. This will favor stations close to the head end. However, a large degree of fairness is achieved by permitting stations to switch from the WAIT to DEFER state only when two consecutive ENDS are encountered for that subcycle. This will only occur when all traffic currently in DEFER state has been granted access. At this point, traffic in the WAIT state would switch to DEFER and then vie for empty slots as they become available. This strategy is related to the snapshot algorithm.[16]

(*iv*) Continuation—In contrast to blocking, continuation requires that traffic not able to access the link in the previous cycle be served before any new traffic is accommodated. This may be achieved in the following manner. Assume that the class type is non-blocking. If the head station should issue a new START before all traffic of the class has been served, the end station will not detect the end of the cycle and hence will not issue END = 1. The absence of an END = 1 would indicate to the head station that the p registers of the stations in that class should not be reset on the next cycle (i.e., the stations would not switch from WAIT to DEFER).[15] Thus, in the following cycle, remaining traffic would be served. For centralized control, the START for this traffic type could contain an additional bit to indicate whether the previous cycle is being continued for deferring traffic or a fresh cycle is being started for new traffic. For distributed control, each station could keep track of the sequence of STARTs and ENDs.

It is important that the control strategy be adaptive to changing traffic conditions. We expect that the traffic mix will change relatively slowly—over a period of seconds rather than ms. Thus, it would be feasible to have the adaptation achieved by a server process.

The control algorithm could be implemented as completely distributed, completely centralized, or somewhere in between. Economics and reliability will dictate, to a large extent, where the control should be placed. Nevertheless, a hybrid strategy would seem more in the spirit of the current design. For example, selection is probably best achieved by having the head station transmit the appropriate START code (centralized, but perhaps assumable), whereas traffic assignment and continuation is probably best achieved by having each station read and operate on the END field (distributed).

VIII. TOPOLOGY

8.1 *Introduction*

A population of stations may be connec
single link (Fig. 9a) or by several interconn
best topology will depend upon physical dist
and the particular performance measures th
We will not consider the general problem, bu
linear interconnection of links forming clos
connected as shown in Fig. 10. Packets in Fa
2 are addressed to station S_N. Station S_N
station S_1 which puts them on Fasnet 2. Si
Fasnet 2 destined for Fasnet 1. To provide
station failures, interconnection stations wc
Thus, a similar connection would be provid
detailed procedure may be specified whereb
nection passes from the $S_N - S_1$ connection t
in case of failure of the former. In principle,
monitors both Fasnets and assumes the inte
a suitable time-out period in event of failu
made for the primary connection to period
ondary connection is operational.

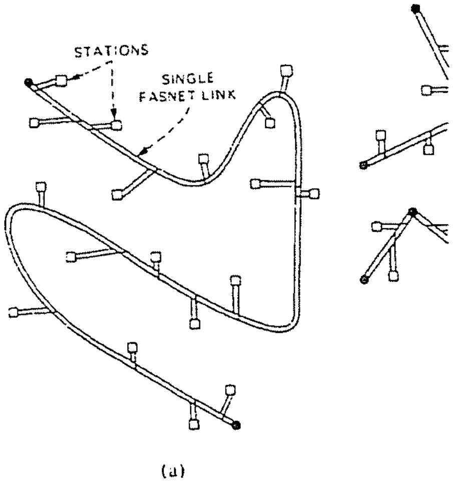


Fig. 9—(a) A cluster of stations being served by a
station population being served by several interconne

ion has a connection.
y have access take the first free slot available to
opriate START. Allocation of freed up slots on a
e basis would proceed as follows. Stations would
its becoming free from the position of the END bit
ote, no END is issued if the class is full. Stations
would be permitted to compete for the n empty
he subcycle. This will favor stations close to the
a large degree of fairness is achieved by permitting
om the WAIT to DEFER state only when two
re encountered for that subcycle. This will only
ic currently in DEFER state has been granted
traffic in the WAIT state would switch to DEFER
pty slots as they become available. This strategy
shot algorithm.[16]
—In contrast to blocking, continuation requires
o access the link in the previous cycle be served
c is accommodated. This may be achieved in the
ssume that the class type is non-blocking. If the
ssue a new START before all traffic of the class
end station will not detect the end of the cycle
ue END = 1. The absence of an END = 1 would
tation that the p registers of the stations in that
set on the next cycle (i.e., the stations would not
DEFER).[15] Thus, in the following cycle, remain-
served. For centralized control, the START for
contain an additional bit to indicate whether the
g continued for deferring traffic or a fresh cycle
ew traffic. For distributed control, each station
e sequence of STARTs and ENDs.
t the control strategy be adaptive to changing
expect that the traffic mix will change relatively
of seconds rather than ms. Thus, it would be
laptation achieved by a server process.
m could be implemented as completely distrib-
ralized, or somewhere in between. Economics
tate, to a large extent, where the control should
ss, a hybrid strategy would seem more in the
esign. For example, selection is probably best
e head station transmit the appropriate START
perhaps assumable), whereas traffic assignment
robably best achieved by having each station
e END field (distributed).

VIII. TOPOLOGY

8.1 Introduction

A population of stations may be connected together by either a single link (Fig. 9a) or by several interconnected links (Fig. 9b). The best topology will depend upon physical distribution, traffic patterns, and the particular performance measures that one seeks to optimize. We will not consider the general problem, but restrict ourselves to the linear interconnection of links forming closed loops. Fasnets may be connected as shown in Fig. 10. Packets in Fasnet 1 destined for Fasnet 2 are addressed to station S_N. Station S_N transmits the packets to station S_1 which puts them on Fasnet 2. Similarly, for packets from Fasnet 2 destined for Fasnet 1. To provide reliability against single-station failures, interconnection stations would be provided in pairs. Thus, a similar connection would be provided between S_{N-1} and S_2. A detailed procedure may be specified whereby control of the interconnection passes from the $S_N - S_1$ connection to the $S_{N-1} - S_2$ connection in case of failure of the former. In principle, the secondary connection monitors both Fasnets and assumes the interconnection function after a suitable time-out period in event of failure. Provision can also be made for the primary connection to periodically check that the secondary connection is operational.

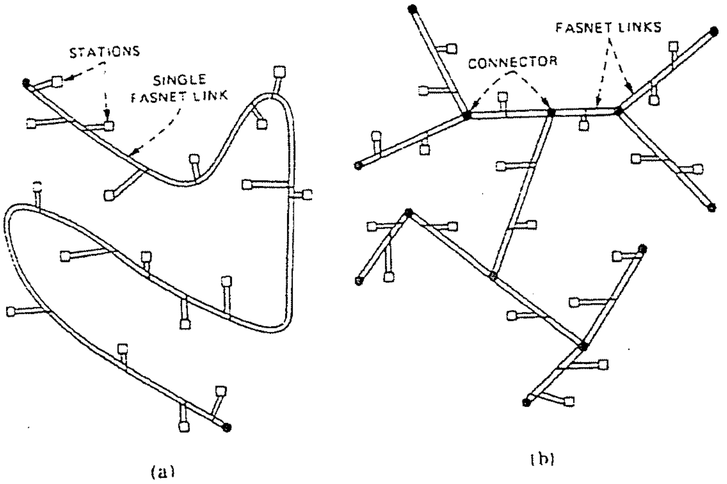


Fig. 9—(a) A cluster of stations being served by a single Fasnet link. (b) The same station population being served by several interconnected links.

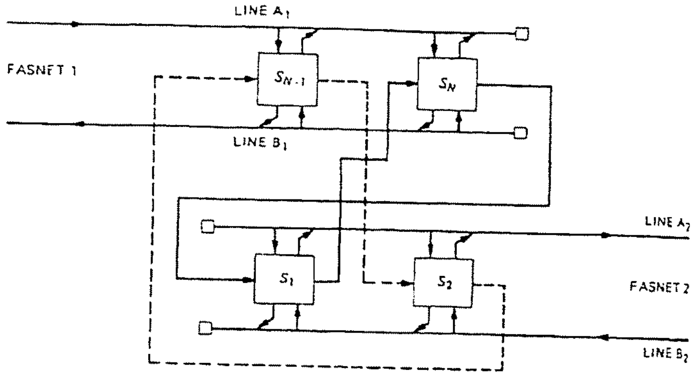


Fig. 10—Structure of a connector used to interconnect two Fasnet links.

Interconnection of Fasnets as shown in Fig. 10 permits traffic to pass from one link to another with a minimum of delay. Since the connection is to the first station on the link, the incoming packet can utilize the next occurring slot. Because of differences in frame timing between two links, it may be necessary to buffer a maximum of one complete packet; this amount of buffering is normally provided in a station interface. In general, interconnection of more than two Fasnets will require larger buffers to be employed to handle the condition where traffic arrives simultaneously for one Fasnet from connecting Fasnets.

8.2 Traffic localization

If stations on a Fasnet have traffic destined only for stations in the immediate vicinity, then total utilization can be significantly improved by dividing the single link into separate links that are connected. Only traffic that has not reached its destination link is allowed to cross the connector.

Consider two Fasnet links connected to form a ring with an inner and outer loop as shown in Fig. 11. The two connectors transfer traffic from one link to the other.

Assume that the ring has a length of unity and that the first link has a length l where $l \leq ½$. For a given packet, let d be the distance of the destination station from the source station. This distance is measured along the ring with the anticlockwise direction as positive, the clockwise direction as negative, and the source station as the origin. Assume that d is a random variable with a uniform distribution over $[-½, ½]$. When $d \epsilon [-½, 0]$, the source station accesses the outer loop; when $d \epsilon [0, ½]$, the source station accesses the inner loop. Thus, the source station selects the shortest distance to the destination along the ring.

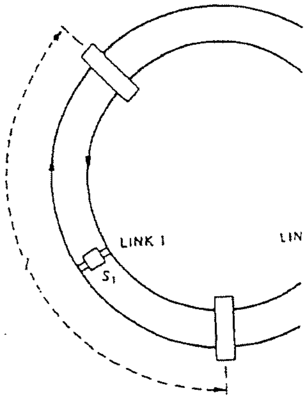


Fig. 11—Two Fasnet links interconnected to form ı

Depending on l and d, the traffic to any traverse zero, one, or two connectors before

We first wish to determine the probabil when the stations are distributed uniformly ı selected at random on link $j\,(j = 1, 2)$, let

p_j = probability that the destination is o same link, and the traffic traverses z

q_j = probability that the destination is c other link, and the traffic traverses (

r_j = probability that the destination is o same link, but the traffic traverses tı if the length of the link is >½ and (two stations at the ends of the link i

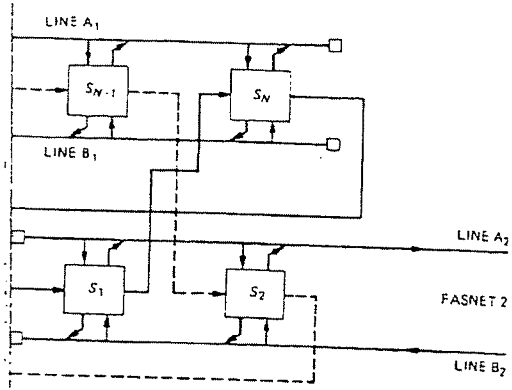
It can easily be shown that

$$p_1 = \frac{l}{2} \qquad p_2 = \frac{1}{2}\left(\frac{3}{4} -\right.$$

$$q_1 = \frac{1-l}{2} \qquad q_2 = \frac{1}{2}\,l$$

$$r_1 = 0 \qquad r_2 = \frac{1}{2}\,\frac{\left(\frac{1}{2} -\right.}{1\cdot}$$

$$\left(\text{as } l < \frac{1}{2}\right)$$



e of a connector used to interconnect two Fasnet links.

Fasnets as shown in Fig. 10 permits traffic to pass
ther with a minimum of delay. Since the connec-
ition on the link, the incoming packet can utilize
it. Because of differences in frame timing between
necessary to buffer a maximum of one complete
of buffering is normally provided in a station
interconnection of more than two Fasnets will
to be employed to handle the condition where
neously for one Fasnet from connecting Fasnets.

net have traffic destined only for stations in the
ien total utilization can be significantly improved
link into separate links that are connected. Only
ached its destination link is allowed to cross the

:t links connected to form a ring with an inner
vn in Fig. 11. The two connectors transfer traffic
ther.

g has a length of unity and that the first link has
. For a given packet, let d be the distance of the
m the source station. This distance is measured
e anticlockwise direction as positive, the clock-
ive, and the source station as the origin. Assume
iable with a uniform distribution over [−½, ½].
source station accesses the outer loop; when
ation accesses the inner loop. Thus, the source
test distance to the destination along the ring.

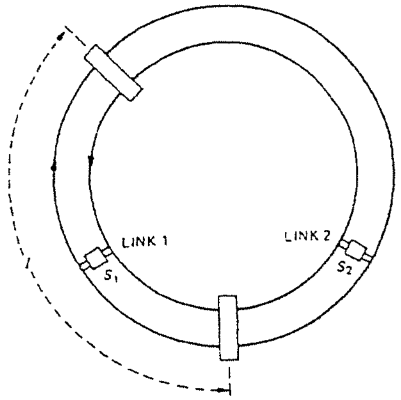


Fig. 11—Two Fasnet links interconnected to form a link using two connectors.

Depending on l and d, the traffic to any destination may have to traverse zero, one, or two connectors before it can be removed.

We first wish to determine the probabilities of the above events when the stations are distributed uniformly on the ring. For station S_j selected at random on link $j\,(j = 1, 2)$, let

p_j = probability that the destination is on the inner loop, and the same link, and the traffic traverses zero connectors,

q_j = probability that the destination is on the inner loop, but the other link, and the traffic traverses one connector,

r_j = probability that the destination is on the inner loop, and the same link, but the traffic traverses two connectors ($r_j \neq 0$ only if the length of the link is >½ and the shortest path between two stations at the ends of the link is through the other link).

It can easily be shown that

$$\begin{aligned} p_1 &= \frac{l}{2} & p_2 &= \frac{1}{2}\left(\frac{3}{4} - l\right)\frac{1}{1-l} \\ q_1 &= \frac{1-l}{2} & q_2 &= \frac{1}{2}\,l \\ r_1 &= 0 & r_2 &= \frac{1}{2}\,\frac{\left(\frac{1}{2} - l\right)^2}{1-l}. \\ &\left(\text{as } l < \frac{1}{2}\right) \end{aligned} \qquad (3)$$

Note that $p_j + q_j + r_j = ½$, $(j = 1, 2)$ as any station accesses the inner loop with probability ½. Now suppose that there are N_1 active stations on link l and N_2 active stations on link 2. If each active station requires unit capacity, then the average traffic $T_i(i = 1, 2)$ in units of capacity on the inner loop in links 1 and 2 is given by

$$\begin{aligned} T_1 &= N_1(p_1 + q_1 + r_1) + N_2(q_2 + r_2) + N_1r_1 \\ T_2 &= N_2(p_2 + q_2 + r_2) + N_1(q_1 + r_1) + N_2r_2 \end{aligned} \tag{4}$$

because the traffic on a line in any link is the sum of three components: (i) all the traffic generated in that link, (ii) that fraction of the traffic generated in the other link that traverses this link, and (iii) that fraction of the traffic generated in this link that transverses the other link and then returns to the first link. If N_1 and N_2 are very large, then by the law of large numbers we have

$$\begin{aligned} \text{traffic on link } 1 &= T_1 \leq C \\ \text{traffic on link } 2 &= T_2 \leq C, \end{aligned} \tag{5}$$

where C is the line capacity.

It can be shown that $N_1 + N_2$ is maximum when $l = ½$ and

$$N_1 + N_2 = \frac{8}{3}C.$$

If no connectors were used, we have a single Fasnet link (it can no longer be a closed ring) for which

$$N_1 + N_2 = 2C.$$

Hence the gain $G = (8C/3)/2C = 4/3$. Thus, we are able to obtain a 33 percent increase in the effective network capacity even for uniformly distributed traffic by having two diametrically located connectors.*

We now extend the above analysis to the case of K connectors C_1, C_2 $\cdots$ C_K, which are located symmetrically around the ring as shown in Fig. 12.

It can be shown by similar means that

$$G = \frac{4}{1 + \frac{4}{K}}. \tag{6}$$

This is plotted in Fig. 13.

We then extend the same analysis to the case with K symmetric connectors C_1, C_2 $\cdots$ C_K, but with an arbitrary traffic distribution symmetric about the source station and extending from −½ to +½

* While this analysis does not pertain to a specific access protocol, the effective gain can be closely realized by the Fasnet protocol and the connector structure of Fig. 10.

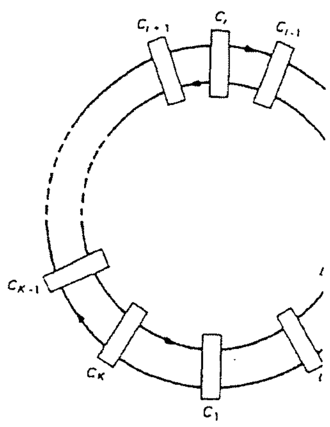


Fig. 12—A Fasnet link with multiple c

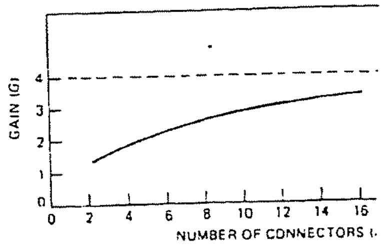


Fig. 13—Plot of the gain in traffic handling capacity c to a single link as a function of K, the number of connec

along the ring. This preserves the shortest sharing the load equally between the two loc

It can be shown by similar means that for

$$G \approx \frac{K}{1 + KD} = \frac{1}{\frac{1}{K} + l}$$

where D is the expected value of $|d|$, the abs destination distance. The approximation be bution is uniform or if $K \rightarrow \infty$. Note that for $D = ¼$ and

$$G = \frac{1}{\frac{1}{K} + \frac{1}{4}} = \frac{4}{1 + 4/}$$

r_j = ½, (j = 1, 2) as any station accesses the inner ½. Now suppose that there are N_1 active stations ve stations on link 2. If each active station requires he average traffic $T_i (i = 1, 2)$ in units of capacity links 1 and 2 is given by

$$
\begin{aligned}
&_1(p_1 + q_1 + r_1) + N_2(q_2 + r_2) + N_1 r_1 \\
&_2(p_2 + q_2 + r_2) + N_1(q_1 + r_1) + N_2 r_2
\end{aligned}
\qquad (4)
$$

a line in any link is the sum of three components: erated in that link, (ii) that fraction of the traffic her link that traverses this link, and (iii) that generated in this link that transverses the other to the first link. If N_1 and N_2 are very large, then umbers we have

$$
\begin{aligned}
&\text{traffic on link } 1 = T_1 \leq C \\
&\text{traffic on link } 2 = T_2 \leq C,
\end{aligned}
\qquad (5)
$$

apacity.

at $N_1 + N_2$ is maximum when l = ½ and

$$N_1 + N_2 = \frac{8}{3} C.$$

e used, we have a single Fasnet link (it can no g) for which

$$N_1 + N_2 = 2C.$$

$8C/3)/2C = 4/3$. Thus, we are able to obtain a 33 ie effective network capacity even for uniformly having two diametrically located connectors.* above analysis to the case of K connectors C_1, C_2 ated symmetrically around the ring as shown in

similar means that

$$G = \frac{4}{1 + \frac{4}{K}}. \qquad (6)$$

13.

e same analysis to the case with K symmetric C_K, but with an arbitrary traffic distribution source station and extending from −½ to +½

* not pertain to a specific access protocol, the effective gain e Fasnet protocol and the connector structure of Fig. 10.

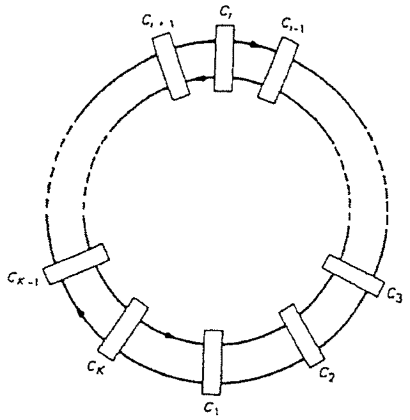


Fig. 12—A Fasnet link with multiple connectors.

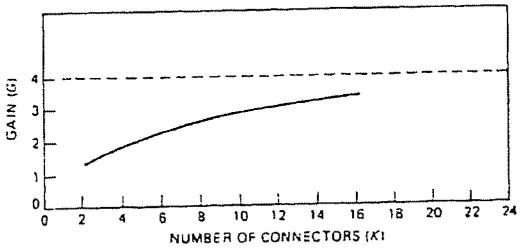


Fig. 13—Plot of the gain in traffic handling capacity of interconnected links relative to a single link as a function of K, the number of connectors.

along the ring. This preserves the shortest path routing, as well as sharing the load equally between the two loops.

It can be shown by similar means that for large K

$$G \approx \frac{K}{1 + KD} = \frac{1}{\frac{1}{K} + D}, \qquad (7)$$

where D is the expected value of $|d|$, the absolute value of the source-destination distance. The approximation becomes exact if the distribution is uniform or if $K \rightarrow \infty$. Note that for the uniform distribution, D = ¼ and

$$G = \frac{1}{\frac{1}{K} + \frac{1}{4}} = \frac{4}{1 + 4/K}$$

as before. For a given traffic distribution, (7) is a good design formula for how capacity costs can be reduced at the cost of extra connectors. However, this trade-off is useful only if accurate estimates of capacity and connector costs are available.

The above analysis highlights certain interesting features. For the uniform traffic distribution, it is seen that the gain, G, does not increase uniformly with K. This is fairly intuitive. Since the traffic is uniform, an extra connector, when K is large, removes the traffic over only a short link and results in only a marginal increase in the gain. On the other hand, as the traffic distribution becomes more localized (i.e., $D \rightarrow 0$)G increases uniformly with K. This is again fairly obvious, as with a high degree of localization the traffic on each link is almost independent of the traffic on the other links.

We have considered here some configurations of interconnected links. There are interesting graph theoretic questions relating to reliability. For example, given a graph like Fig. 9b, what is the minimum number of additional links and their position so that full connectivity is still maintained if any link is cut at a single point? Development of realistic models of the physical traffic and cost structures of local environments still remains. There is a paucity of statistics (except for Ref. 2) on the local network parameters. Future operational local networks will hopefully furnish statistics on which to build more accurate models.

IX. CONCLUSION

The physical configuration of Fasnet consists of two communication lines passing each station. One line carries traffic in one direction, while the other line carries traffic in the opposite direction. Thus, this configuration carries twice the traffic of a previous system in which the two lines were connected at one end so that traffic was written on the outbound line and read from the inbound line. Each station makes two connections to each line, a nondirectional read tap and a directional write tap. Reliability of the physical medium is high because it contains no active electronics. The access protocol is partly centralized in that bit synchronization, framing, and start-of-cycle are provided by the end stations; however, these functions would be assumable by any station upon failure of an end station.

The access protocol is as follows: Upon reading a start-of-cycle, a station may transmit a prespecified number of packets in the first available empty slots. When all stations have transmitted their packets, a signal is sent on the return line to inform the head station to start a new cycle. The efficiency of Fasnet increases as the length of a cycle increases; cycle length depends upon the length of a packet, the number of active stations, and the number of packets, p_{max}, that each station is permitted to send in a cycle. By
efficiency can be maintained at a high level
of active stations. A number of techniqu
efficiency are suggested. A trade-off is nece
the complexity of the protocol, on one han
improvements in efficiency on the other.

Bit synchronization of the stations is acl
out-of-band pilot tone, while framing is achic
inserted code word. A three-level bipolar lir

The potential of Fasnet for operation a
makes it attractive as a conduit for the va
may flow in a business environment. Mecha
to implement blocking, delaying, and requ
that are needed if mixed traffic is to be ha
single medium. These operations can be imp
can be distributed. The low-level access ope
while the more complex operations are best

Fasnets may be interconnected to incre
carried or to improve reliability. Investigat
the connection of Fasnets to form a ring. A
in the ring increases, the throughput first ir
uniformly distributed traffic). After about f
is very small. Exploration of other topologi

X. ACKNOWLEDGMENT

We thank B. Gopinath and N. F. Maxen
and Z. L. Budrikis of the University of We
discussions during various stages of this wo

REFERENCES

1. R. M. Metcalfe and D. R. Boggs, "Ethernet: Distri
Computer Networks," Commun. of the ACM *19*
2. J. F. Shoch and J. A. Hupp, "Measured Performan
Presented at Local Area Communications Ne
1979), pp. 113–34.
3. The Ethernet, A Local Area Network, Data Link
fications, DEC, INTEL, XEROX, Version 1.0 (S
4. F. A. Tobagi, "Multiaccess Protocols in Packet
Trans. Commun. *COM-28*, 4 (April 1980), pp. 4
5. J. O. Limb, "High Speed Operation of Broadcast I
1982.
6. M. E. Ulug, G. M. White, and W. L. Adams, "B
Proc. Seventh Data Commun. Symp. (October
7. M. V. Wilkes and D. J. Wheeler, "The Cambrid
Proc. Local Area Commun. Network Symp. (M.
8. C. S. Skrzypczak and W. E. Falconer, "Bell Syste
June 1981, pp. 19.6.1–6.
9. N. F. Maxemchuk and H. A. Wilder, "Virtual
Information Systems Workshop, St. Maximim,

en traffic distribution, (7) is a good design formula
sts can be reduced at the cost of extra connectors.
-off is useful only if accurate estimates of capacity
are available.

is highlights certain interesting features. For the
bution, it is seen that the gain, G, does not increase
his is fairly intuitive. Since the traffic is uniform,
when K is large, removes the traffic over only a
ts in only a marginal increase in the gain. On the
traffic distribution becomes more localized (i.e.,
uniformly with K. This is again fairly obvious, as
of localization the traffic on each link is almost
raffic on the other links.

red here some configurations of interconnected
resting graph theoretic questions relating to reli-
given a graph like Fig. 9b, what is the minimum
l links and their position so that full connectivity
any link is cut at a single point? Development of
the physical traffic and cost structures of local
mains. There is a paucity of statistics (except for
l network parameters. Future operational local
ully furnish statistics on which to build more

guration of Fasnet consists of two communication
ation. One line carries traffic in one direction,
arries traffic in the opposite direction. Thus, this
twice the traffic of a previous system in which
nnected at one end so that traffic was written on
l read from the inbound line. Each station makes
ach line, a nondirectional read tap and a direc-
ability of the physical medium is high because it
ctronics. The access protocol is partly centralized
ation, framing, and start-of-cycle are provided by
ever, these functions would be assumable by any
f an end station.

l is as follows: Upon reading a start-of-cycle, a
a prespecified number of packets in the first
When all stations have transmitted their pack-
n the return line to inform the head station to
efficiency of Fasnet increases as the length of a
ength depends upon the length of a packet, the
ons, and the number of packets, p_{max}, that each station is permitted to send in a cycle. By adaptively changing p_{max}, efficiency can be maintained at a high level even for a small number of active stations. A number of techniques for further improving efficiency are suggested. A trade-off is necessary between increasing the complexity of the protocol, on one hand, and the resulting small improvements in efficiency on the other.

Bit synchronization of the stations is achieved through adding an out-of-band pilot tone, while framing is achieved through a periodically inserted code word. A three-level bipolar line code is preferred.

The potential of Fasnet for operation at high transmission rates makes it attractive as a conduit for the various types of traffic that may flow in a business environment. Mechanisms have been proposed to implement blocking, delaying, and request-for-service operations that are needed if mixed traffic is to be handled efficiently within a single medium. These operations can be implemented centrally or they can be distributed. The low-level access operations are best distributed while the more complex operations are best centralized.

Fasnets may be interconnected to increase the load that may be carried or to improve reliability. Investigation has been restricted to the connection of Fasnets to form a ring. As the number of segments in the ring increases, the throughput first increases rapidly (assuming uniformly distributed traffic). After about five segments, the increase is very small. Exploration of other topologies presents a challenge.

X. ACKNOWLEDGMENT

We thank B. Gopinath and N. F. Maxemchuk of Bell Laboratories and Z. L. Budrikis of the University of Western Australia for helpful discussions during various stages of this work.

REFERENCES

1. R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," Commun. of the ACM *19*, 7 (July 1976), pp. 395–404.
2. J. F. Shoch and J. A. Hupp, "Measured Performance of an Ethernet Local Network," Presented at Local Area Communications Network Symposium, Boston (May 1979), pp. 113–34.
3. The Ethernet, A Local Area Network, Data Link Layer and Physical Layer Specifications, DEC, INTEL, XEROX, Version 1.0 (September 30, 1980).
4. F. A. Tobagi, "Multiaccess Protocols in Packet Communication Systems," IEEE Trans. Commun. *COM-28*, 4 (April 1980), pp. 468–88.
5. J. O. Limb, "High Speed Operation of Broadcast Local Networks," Proc. ICC, June 1982.
6. M. E. Ulug, G. M. White, and W. L. Adams, "Bidirectional Token Flow System, Proc. Seventh Data Commun. Symp. (October 1981), pp. 149–55.
7. M. V. Wilkes and D. J. Wheeler, "The Cambridge Digital Communication Ring," Proc. Local Area Commun. Network Symp. (May 1979), pp. 47–61.
8. C. S. Skrzypczak and W. E. Falconer, "Bell System Planning of ISDN," Proc. ICC, June 1981, pp. 19.6.1–6.
9. N. F. Maxemchuk and H. A. Wilder, "Virtual Editing: I. The Concept," Office Information Systems Workshop, St. Maximim, France, October 1981.

10. W. Pferd, L. A. Peralta, and F. X. Prendergast, "Inactive Graphics Teleconferencing," Computer *12* (November 1979), pp. 62–71.
11. W. A. Rheinfelder, *CATV System Engineering*, Third Ed., Blue Ridge Summit, Pa.: TAB Books, 1972.
12. J. O. Limb, "Fasnet: Proposal for a High Speed Local Network," Office Information Systems Workshop, St. Maximim, France, October 1981.
13. IEEE 802 Standards Committee, Interim Report.
14. F. D. Waldhauer, "A 2-Level, 274 Mb/s Regenerative Repeater for T4M," Proc. of the ICC 1975, *3*, pp. 48-13–17.
15. L. Frata, F. Borgonovo, and F. A. Tobagi, "The Express-Net: A Local Area Communication Network Integrating Voice and Data," in *Performance of Data Communication Systems*, G. Pujolle, Ed., Amsterdam: North Holland, 1981, pp. 77–88.
16. R. C. Chen, "Bus Communication Systems," Report No. PB 235-897, Carnegie-Mellon University, January 1974.

Copyright © 1982 American Telephone and Telegraph Compan
THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 61, No. 7, September 1982
Printed in U.S.A.

On the Distribution Function
Power Sums With Log-Norm

By S. C. SCHWARTZ* and

(Manuscript received July 2

An approximate technique is presented
mean and variance of the power sums wit
Exact expressions for the moments with t
oped and then used in a nested fashion to
desired sum. The results indicate more a
quantities over a wider range of individ
than any previously reported procedure. C
the Gaussian assumption for the power s
zation of the cumulative distribution func
ably well with a Monte Carlo simulation i
of the variate. Simple polynomial express
ments lead to an effective analytical tool f
ance studies. They allow quick and accura
such as cochannel interference caused b
lephony.

I. INTRODUCTION

The power sum with K independent com

$$P_K = 10 \log_{10}\left[\sum_{k=1}^{K} 10 \right.$$

is a random variable which appears in man
With X_k Gaussian, the quantity

$$L_k = 10^{X_k/10}$$

is called a log-normal variate. The charact

[*] Professor Schwartz is currently with the Depart
Computer Science, Princeton University.

EXHIBIT 10

October 1984
Vol. 63 No. 8 Part 1

AT&T BELL LABORATORIES TECHNICAL JOURNAL

A JOURNAL OF THE AT&T COMPANIES

QAM Binary Signaling

Digital Radio

Speech Processing

Packet/Circuit Switching

Traffic

AT&T Bell Laboratories Technical Journal
Vol. 63, No. 8, October 1984
Printed in U.S.A.

A Packet/Circuit Switch

By Z. L. BUDRIKIS* and A. N. NETRAVALI†

(Manuscript received July 29, 1983)

We propose a switch, suitable for an integrated local communications network, that will support packet switching and circuit switching, with a wide range of bit rates. Key components are two serial memories; a multiplicity of access units, each capable of writing and reading uniformly formatted, addressed information; and a programmed controller. Circuit switching is achieved when the controller repeatedly allocates memory slots, following call setup. Data communications can proceed concurrently without setup, competing for unused slots. We give an example of a 10,000-telephone-line switch carrying a similar load of other traffic. The switch would delay voice by less than 5 ms and could be interfaced to the existing telephone system. We indicate a method of fault detection and isolation that will limit the impact of a failure on a serial memory to an arbitrarily small group of connected lines. We define an index for measuring failure impact and use it to derive most-favorable fault-isolating partitions.

I. INTRODUCTION

The telephone system is by far the world's largest communications network. It was primarily designed for voice, but its role widens continuously, as it adapts to new requirements. Presently it is changing to accommodate data communications.

Already the network extensively caters to data communications, but not yet as well as it might. Although internally the telephone system

* AT&T Bell Laboratories. On leave from the Department of Electrical and Electronic Engineering, University of Western Australia, Nedlands. † AT&T Bell Laboratories.

is rapidly becoming a vast interconnected computer system, proffered data are still largely carried as analog signals externally. That will change, however, as special provisions for data come on-line. As more of the plant, including switches, becomes digital, it will be possible to offer, on a selective basis, switched digital telephone channels usable for 56-kb/s data throughput. Also, packet-switched data services will widen in scope and access. Packet-switched data services are overlay networks that use the digital transmission facilities of the telephone network but bypass its switches, of which many are still analog. The packet networks eventually may become totally interconnected, just as the voice network, and also may become integrated with it.

In-house, or proprietary, telephone networks can benefit from the changing character of the overall network more immediately. Already available are switches and other components that permit an all-digital network that will accommodate on one facility both voice and data. As good as this already is, we are proposing a switch that could make the private network even better. Eventually it might even influence the entire system.

Currently available switches provide only circuit-switched connections. This gives fixed-capacity channels on a continuous basis, whereas much of data comes in bursts. Thus, computer communications are characterized by very long call durations with only low average, but in many instances very high, peak rates. Given the option, direct memory transfers could proceed in some instances at rates of many megabits per second. This is far too high for a switched and continuously held circuit.

It is true that the needs of bursty traffic can be catered to by what already is available, namely by some packet-switched networks. But that introduces a separate communications network for data, with the consequences of proliferating wiring plans, divided responsibilities, and probable long-term dyseconomies. It is better for one facility to serve all communications, and to do so without imposing mismatches.

We propose a switch and, more generally, a new switch architecture that support within one switching fabric both circuit- and packet-switched connections. This would largely avoid mismatches in respect to bursty data traffic, while preserving unity in communications.

The cardinal components of the switch (see Fig. 1) are a pair of Serial Memories (SMs), a Central Controller (CC), and Accessing Units (AUs). The memories do not recirculate and both ends (head and tail) of each terminate on the central controller. The AUs are connected to read-and-write taps along the SMs, an AU having one connecting tap to each memory. The two taps of an AU form a symmetrical pair: the tap to the second memory is as many places from the tail end as that to the first is from the head. Thus, each AU

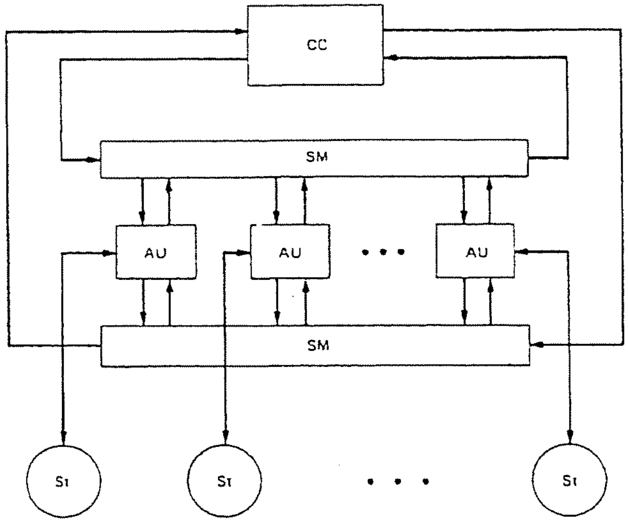


Fig. 1—Block schematic of switch. Access Units (AUs) communicate via two Serial Memories (SMs) on behalf of client Stations (Sts). Central Controller (CC) reserves slots for circuit-switched communications.

can reach every other AU by either one, or the other, memory. It can reach, and be reached by, the central controller by either memory. All writing is logical OR.

An AU acts as an agent of a client station (St) (e.g., telephone, facsimile terminal, computer) and mediates communications between it and other stations by way of corresponding AUs. Communications are carried on by write-and-reads in memory/time slots of uniform length and format. Each slot consists of a data field and several control fields. Collectively, the control fields provide synchronization, "Slot busy" indication, source and destination addressing, and slot pleading. A circuit-switched communication is carried on in regularly recurring slots, which are appropriately premarked by the central controller. For a packet-switched communication an AU simply uses the next available slot.

The capacity of a connected circuit can vary over a wide range, from a small fraction of a single (64-kb/s) telephone channel up to a large multiple of that capacity. It is settled by negotiation with the CC at the time of circuit setup, and need not be the same on different occasions. The capacity available to a packet-switched communication depends on the prevailing competition and can be any portion of the total switch capacity. The latter is a function of size and would be just several megabits per second for a 100-line switch and several hundred megabits per second for a switch that supports 10,000 lines.

A simple realization of the serial memories would be by clocked shift registers. The shift registers can be bit-paralleled to any degree needed to keep the clock rate low. The memories and all access units can be located centrally at the controller, with all connections to the switch then forming a single star. But it is also possible to segment the memories and form the network in clusters. The segments of memory would be connected serially to each other and to the central controller by transmission lines, forming two contrary rings, and the clusters again would form star topologies.

All elements of our proposal are well established and tried. Central, or stored program, control in circuit switching is over twenty years old.[1] The idea of switching by time slot interchanges is even older,[2] followed shortly by its realization through read-and-writes in computer memory.[3,4] Packet switching is more recent,[5] but is also well established both in local and wide-area networks.[6–9]

In essence, our scheme is an adaptation of seemingly diverse procedures, so that they may coexist. Time division slots are enlarged from what is usual in circuit switching, so that they can carry the control information essential to packet switching. Unlike normal packet-switched schemes, packets are of a single fixed length so that they can also be circuit-switched. Instead of separate time and space division stages, common in current telephone switches, we have a combined space/time fabric, abstracted from ring and bus networks, with a particular debt to Fasnet.[9] This makes packet switching possible without controller intervention. Finally, the controller maintains circuit connections by repetitive slot allocations, which is only marginally different from what takes place in a time division stage of a standard switch.

Also, our proposal is not first in its suggestion that voice and data be integrated on a common network.[10–14] But it appears to be first in suggesting a common switch for circuits and packets as the basis for that integration. With few exceptions,[11] prior suggestions have been to treat voice as data and to packet-switch it both in local and wide-area networks. However, these proposals have attendant delays that have to be addressed.

The point is important since, in the global telephone system, transmission delays can limit the quality of many possible connections. In the case of our switch, the delay of voice signals can be kept to less than 5 ms. It depends only on the clock rate, the size of switch, and the size of slots. Since delay considerations have an overriding sway on system choices, we discuss them in Section II. In Section III we give further details of our proposal.

In Section IV we address the question of reliability in our switch. We do this because our proposal may be seen as being particularly vulnerable, since all its communications are to take place via two serial memories to which all AUs have writing privileges. We introduce a scheme for sectional detection and isolation of faults applicable to our switch. We show that this would limit the impacts of faults in our case to those that would prevail in switches that have much more dispersed and/or redundant architectures.

II. DELAY CONSTRAINTS AND RATE REQUIREMENTS

A communication system is expected to deliver messages to the destination in a timely fashion. The permitted delay is different in character for data and for real-time signals. We review the two cases separately.

2.1 *Data transmission*

Within limits, the exact times of arrival at the destination of the different parts in a data stream generally are unimportant. It is usually required that the sequence in the stream be preserved and that the average delay does not exceed some specified value. When data are presented for transmission at a fluctuating rate and there is not sufficient transmission capacity to cope with the peaks, the flow is smoothed by buffering. Waiting times in buffer stores are the predominant cause of delay.[15]

2.2 *Real-time signals*

In the transmission of real-time signals, the delay should be a constant and not greater than a specified value. Given fluctuation in transmission rate, there will be a time-varying delay $W_s(t)$ in a buffer at the sending end. A further delay $W_r(t)$ must be deliberately introduced in a buffer at the receiver,[16] so that the total delay could stay constant:

$$W_s + W_r = D. \tag{1}$$

D is the fixed buffer delay with which the system has been designed.

If, at some time, W_s exceeds D, then, at the same time, the buffer at the receiver will become empty and there will be a break in the received signal. Hence, there is no point in storing more at the transmitter than the amount of data that represents the total designed delay. If the rate λ of the real-time data is constant, as in Pulse Code Modulation (PCM) voice, then waiting times are directly related to amounts of stored data. The buffer-store capacities, N_s and N_r, that need to be provided at the two ends are equal, given by

$$N_r = N_s = \lambda D. \tag{2}$$

If λ is not constant, then the required capacities are still equal and are found by substituting the maximum value of λ in eq. (2).

It is important to note that, given fluctuations in data and/or transmission rate(s) and buffer stores to smooth them, the relevant delay for real-time signals is the maximum, i.e., designed, value, not the statistical average. How much larger that designed delay is to be than the average depends on the actual fluctuations in rate(s) and the relative tolerance to lost quality by signal discontinuities and by delay.

III. DESCRIPTION OF SWITCH

We now detail several aspects of the proposed switch. We give an indication of architectural options, describe protocols, and suggest suitable parameters for a 10,000-line switch.

3.1 Architecture

The basic configuration of the switch was shown in Fig. 1 and outlined in the Introduction. The functions of the AUs and the CC will be defined in more detail when we discuss protocols in the next subsection. It will be seen that there are considerable differences in the tasks of an AU that is mediating a circuit-switched, as compared to packet-switched, communication. Further differences in speed and buffer requirements may be identified between, and within, those two categories.

Clearly, there is a choice between designing a number of special-purpose AUs and designing a single universal AU. Further choices concern sharing of, and multitasking by, access units. Should AUs be placed in a common pool and shared by a larger group of stations? That would entail further switching outside the main switch to mediate connections between AUs and stations. Should an AU be multitasked, serving simultaneously different stations? That would make the AU a more complex device. Figure 2 illustrates a switch that incorporates both sharing and multitasking.

Our inclination is towards universal AUs, one to each station, and towards neither sharing nor multitasking. True, this calls for the largest number of AUs, and not the least complex, at that. But it has the advantage of uniformity and, in the light of technology trends, of likely overall economy.

The next choice concerns the serial memories. They may be active, made in semiconductor, or also, reverting to earlier technologies, passive, e.g., acoustic or electromagnetic delay lines. Passive components are attractive because they promise more reliability. However, our purpose would be better served by clocked shift register memories in an arrangement as, say, shown in Fig. 3. This makes for easier synchronization and permits bit-paralleling to hold down clock rates.

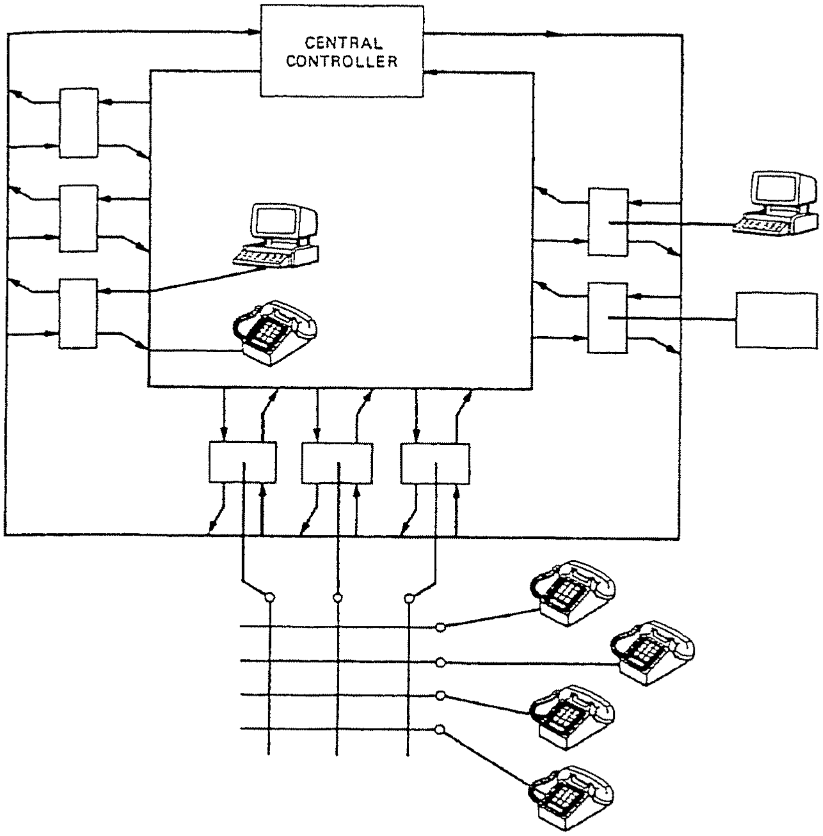


Fig. 2—Different options in AU tasking. All AUs could be of one type, each serving a single station (right); AUs could be shared by a larger group of stations requiring selector switching outside the main switch (center); or an AU could be multitasked, serving more than one type of station (left).

Reliability is a matter of overall design and implementation. In Section IV we discuss an architecture-related aspect of reliability, namely isolation of faults to limited sections.

Finally, we have the question of overall network topology. Three different arrangements are shown in Fig. 4. Figure 4a shows the traditional topology of a central switch and star network. In Fig. 4b a completely distributed arrangement is shown in which the serial memories wend their way past every station. This would make it similar to a local area ring network and would be possible only with passive lines as the memories. A compromise between the above two, and an interesting topology for a PBX that has to serve an extended area, is shown in Fig. 4c. The serial memories are cut into sections, and each section is placed close to the group of stations that it serves.

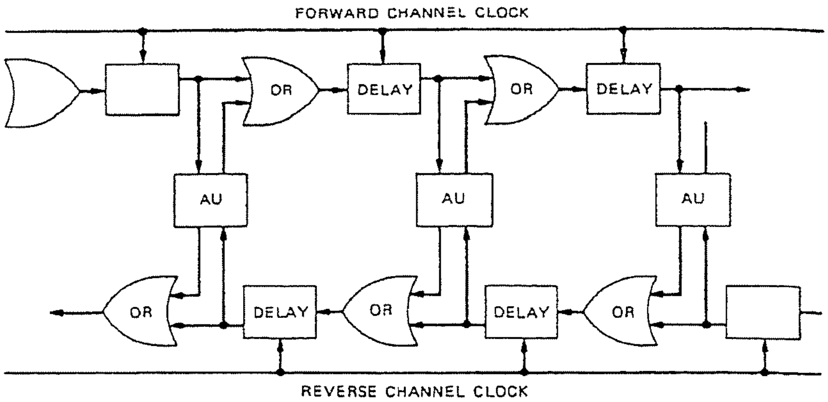


Fig. 3—Shift register realization of serial memories. Access points are at the inputs of clocked unit delays; the writing is through OR gates.

The lines connecting the individual sections and the central controller could be optic fibers, which would carry the total information streams serially even in a large switch.

3.2 Protocols

In the context of our proposal, a packet used in packet-switched communication is made up of five control fields and data, as shown in Fig. 5. The same format could be used in circuit-switched packets. But for these, at least one of the two address fields is unnecessary. Its space may either be added to the data field, or it could be used as a separate channel, a companion to the main channel.

The six fields marked in Fig. 5 are:

1. BUSY—a single bit to indicate slot occupancy
2. RQST—a single bit, common channel used for slot pleading
3. SNDR—address or password of AU sending packet
4. RCVR—address or password of AU intended to receive packet
5. DATA—data field
6. SYNC—synchronization field.

The roles of all the fields, except RQST and SYNC, are self-evident. RQST is used by packet-switching AUs and we will see its function presently when we discuss data communications. The SYNC field is written by the central controller to ensure slot and frame synchronization. Although both synchronizations could be achieved with just one bit per slot, a field of two bits will make them more secure. Altogether, the following numbers would be of the right order: BUSY and RQST one bit each, SYNC two bits, the addresses 14 bits each, and DATA 192 bits, for a total packet of 224 bits, or 28 bytes.

Corresponding to a packet, one may think of a time slot and of a

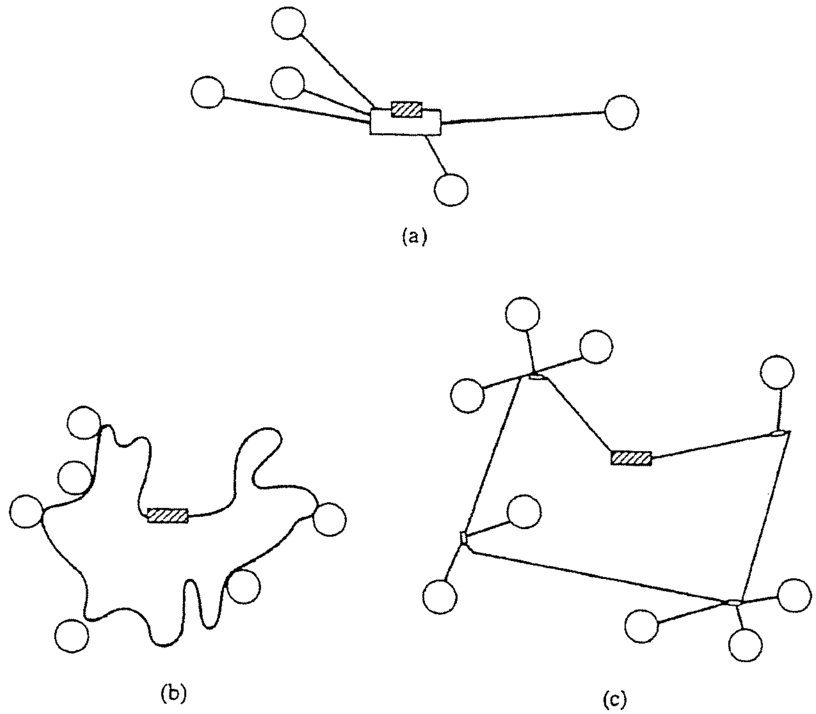


Fig. 4—Network topology options: (a) central switch, stations connected by lines; (b) switch completely distributed with AUs at individual stations and connected by the serial memories realized as buses; (c) switch distributed in clusters, the seral memories within clusters realized by shift registers and between clusters by transmission lines.

propagated memory block as consisting of the same number of bits and divided among respective fields. Note, however, that it is not necessary that the different parts of a packet be placed into a single slot or block. There may be interleaving of packet parts to any extent that is desirable.

Thus, it is conceivable that in order to alleviate the pressure of time for the signaling from receiver to transmitter within the AU, the

BUSY	RQST	SNDR	RCVR	DATA	SYNC

BUSY – SLOT BUSY FIELD
RQST – SLOT REQUEST FIELD
SNDR – SENDER ADDRESS
RCVR – RECEIVER ADDRESS
DATA – DATA FIELD
SYNC – SYNCHRONIZATION FIELD

Fig. 5—Slot format. Typically, BUSY, RQST and SYNC would be one-bit fields, the address fields could be two bytes each and the data field 24 bytes.

BUSY field could refer to the state of occupancy—not of the slot that it is riding in, but of the following slot. Similarly, other fields could be advanced or retarded, and not necessarily by only one slot, nor, indeed, just as a complete field. Thus, the DATA field could be broken into single bytes or even bits, and the fragments made to follow the header as an arbitrarily dispersed tail, provided only that all packets are fragmented and dispersed identically.

The extreme fragmentation of packets, as just alluded to, may seem an attractive way of restoring smoothness to data flow for circuit-switched communications. Indeed, almost complete smoothness is possible for any one chosen rate. But it would be at the expense of considerable complication for all other communications, particularly the packet-switched and the circuit-switched that have higher rates than the one singled out for favorable treatment. We will dismiss it from further consideration and turn to describing procedures.

3.2.1 Data communication

Assume that AU addresses are in numerical order along the two memories, ascending in the direction of propagation along one and descending along the other. We will call the memory with ascending addresses the forward channel, and hence the other the reverse channel.

Suppose that an AU has to communicate to another AU of higher address. It must send a message, or packet(s), on the forward channel. To do so, the dispatch processor of the AU will follow the data dispatch routine of Fig. 6. This can be understood more easily with the help of the state diagram of Fig. 7. For the sake of description, this diagram relates to an exclusive forward channel dispatcher, although in practice a single dispatcher would service both directions.

When idle, the dispatcher is normally in the "Go" state and monitors the sending buffer (for the forward channel), checking whether it contains a packet for transmission. If it does, it reads the BUSY field of the next block on the forward channel and at the same time writes a "ONE" in that field so as to seize the slot, should it be available. If it is not, i.e., BUSY was already "ONE," then it will write "ONE" in the next RQST field on the reverse channel and wait for the next BUSY field on the forward channel. It will repeat reading and writing of BUSY on the forward channel and sending RQSTs on the reverse channel until a "ZERO" BUSY occurs. It will then write in the related SNDR, RCVR, and DATA fields, so dispatching a packet.

Having sent a packet, the dispatcher moves to the 'One packet sent' state. If the sending buffer has at that moment one or more further packets for dispatch, then the dispatcher will behave exactly as in the "Go" state and send off the next packet, thereby moving to the "Two

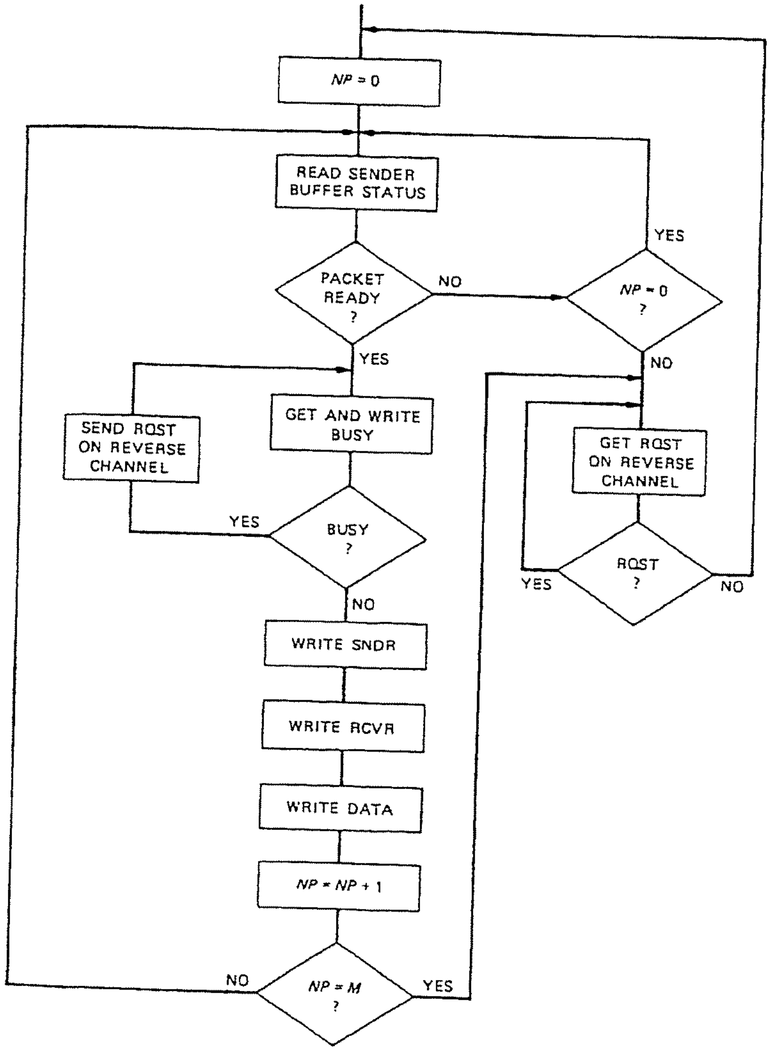


Fig. 6—Flowchart of forward channel data dispatch routine.

packets sent" state. But if there is no packet in the sending buffer on entry to the "One packet sent" state, then the dispatcher will proceed to the "Halt" state. It will remain there until the next "ZERO" is written in the RQST fields on the reverse channel, whereupon it will revert to the "Go" state. Similar conditions apply on entry to the "Two packets sent" and further states, until the dispatcher has sent in a contiguous sequence *M* packets and entered the "*M* packets sent" state. From this it must proceed unconditionally to "Halt."

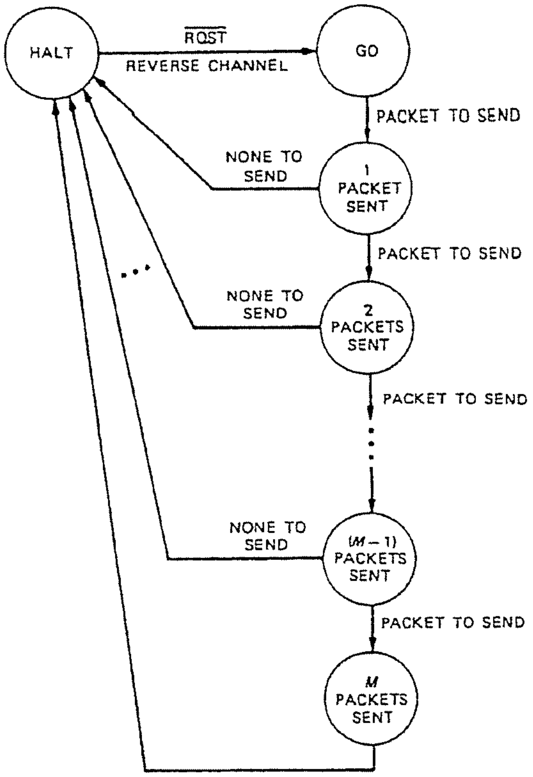


Fig. 7—State diagram of data dispatcher. The dispatcher goes temporarily into HALT state whenever it has no more packets to send or has already sent M packets since the last HALT. It goes from HALT to GO as soon as the RQST bit on the reverse channel is ZERO.

M is a parameter that may vary with AU. It represents priority standing: The larger its value, the less sensitive the AU is to pleadings for slots by other AUs that are downstream from it. It is normally set in relation to the rate of the station that the AU serves.

The task of receiving is less involved but no less time consuming, and an AU will have a separate processor for it. A routine that it could follow is given in Fig. 8. This is set out on the assumption that the SNDR and RCVR fields of a packet would precede the DATA field by one slot.

3.2.2 *Real-time signal transmission*

An AU serving a real-time device has to act in two distinct modes, one in setting up or tearing down a circuit and the other in transmitting and receiving the real-time signals when the circuit is set up. We

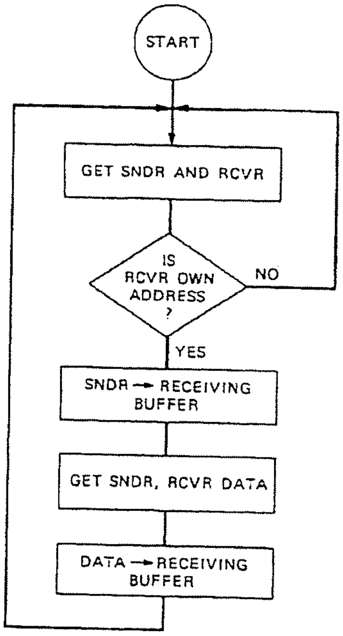


Fig. 8—Flowchart of data receiver routine.

outline the procedure, limiting our attention to telephony. Other devices requiring circuit connections would be served similarly.

Looked at from the telphone, the AU would appear as the line selector of the standard switch. When the telephone is taken off hook, the AU would supply dial tone. As the number is dialed, it would be stored by the AU, which, on completion, would assemble a packet for transmission to the CC. The DATA field of that packet would disclose the fact that a telephone link is being sought, and the numbers of the calling and called stations. The sending procedure for the packet could follow the routine of Fig. 6, even though a simpler routine is possible since no "Halt" state is necessary.

The CC would process received requests using a routine that could be as in Fig. 9. First, the CC would check the total switch capacity already committed to circuit traffic, and from this it would decide whether the setting up of the further circuit is permitted. If it is not permitted, then the CC would inform the originating AU, and that would terminate the processing. If setting up the circuit is permitted, then the CC would determine which AU serves the called station and check whether it is engaged. If it is engaged, then the CC would inform the originating AU accordingly. If it is not engaged, then the CC would tag both AUs as engaged and send messages to both AUs and inform

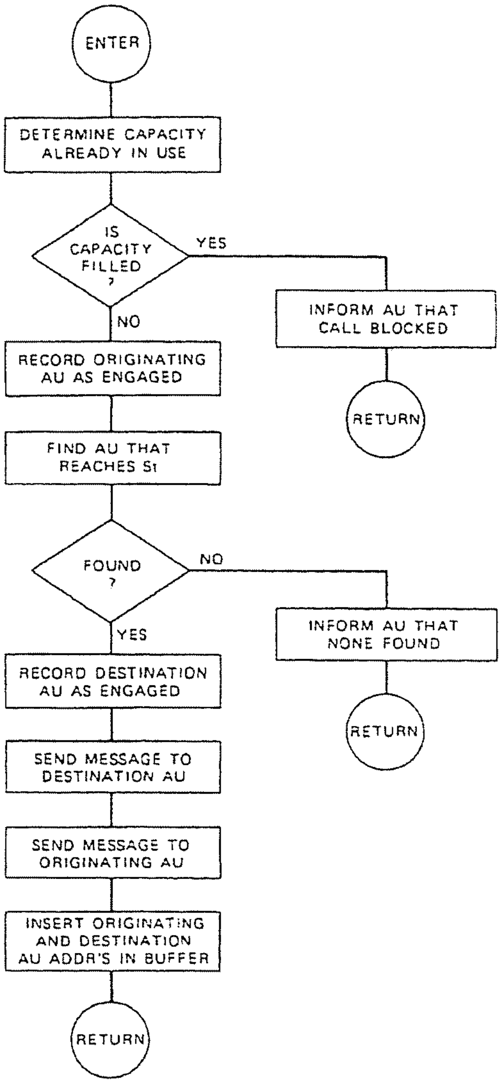

ENTER
DETERMINE CAPACITY ALREADY IN USE
IS CAPACITY FILLED ?
YES
INFORM AU THAT CALL BLOCKED
RETURN
NO
RECORD ORIGINATING AU AS ENGAGED
FIND AU THAT REACHES S1
FOUND ?
NO
INFORM AU THAT NONE FOUND
RETURN
YES
RECORD DESTINATION AU AS ENGAGED
SEND MESSAGE TO DESTINATION AU
SEND MESSAGE TO ORIGINATING AU
INSERT ORIGINATING AND DESTINATION AU ADDR'S IN BUFFER
RETURN

Fig. 9—Flowchart of circuit setup routine in central controller.

them of each other's addresses. The two addresses would also be inserted at appropriate places in ring buffers to cause the necessary premarking of slots by writing of BUSY and SNDR on the correct channels at the right frequencies. This would complete the setting up of the two-way circuit. Given a setup circuit, the dispatch and reception of the real-time data would follow the routines of Fig. 10.

Note that only the SNDR address is used in circuit-switched trans-

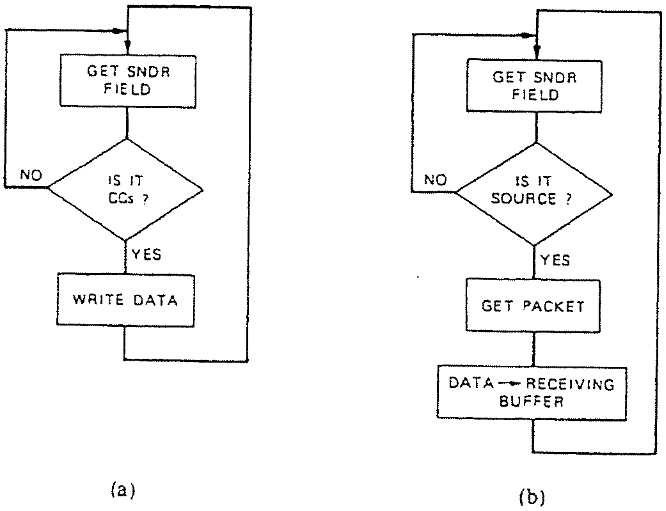


Fig. 10—Flowchart of (a) real-time signal dispatch, and (b) real-time signal reception.

missions: The receiver is given the sender's address and recognizes it for the duration of the call. Apart from saving one address field for other use, there is a further bonus in that more than one receiver can be given the same SNDR address and simultaneously receive the same real-time signal. This leads to the possibility of a simple arrangement for broadcasting to designated outlets for, say, a public address system. If, furthermore, the AU's receiver capability was enlarged to noting several SNDR addresses and taking in packets with those markings, then a telephone conference facility, with voice signal summation at each receiver, would be possible.

The setting up, tearing down, and maintaining of calls to subscribers outside the switch's own area would have to interwork with equipment in other offices. But there is no particular problem about this. The AU serving a trunk would interface with the outside system, sending and responding to signals in conformity with existing specifications. But, in other respects, it would not be different from an AU serving a local subscriber. Data out of, and into, the local switch area could also be carried by circuit-switched trunks, with suitable interfacing to a wider-area data network. The role of an AU providing that interfacing would then amount to that of a gateway processor.

3.3 *Packet size and clock rates*

The bit rate required along the SMs is related to the total peak load for which the system is designed, multiplied by a factor that accounts for efficiency. Assuming a telephone voice signal sampled at 8 kHz, represented by 8 bits per sample and an allowed delay due to packetization of 3 ms, a packet may contain 24 samples or 192 bits of data. The overheads are mainly in the addresses: Assuming a 10,000-voice-line switch and a total number of AUs not exceeding 16K, the SNDR and RCVR fields could be 14 bits each. As we already noted, BUSY and RQST need be only 1 bit each, and SYNC 2 bits. The total overhead will then be 32 bits and the packet length will be 224 bits.

Another criterion by which the overall size of a packet can be decided is efficiency. Since the allowable delay for voice is binding, the best size indicated for maximum efficiency will be of interest only if it is smaller than that already decided.

It is a reasonable simplification to suppose that all offered traffic divides into two categories: very short bursts, and prolonged streams. Furthermore, it is reasonable to assume that the number of packets-per-second from the very short burst will be independent of packet size. Thus, such very short bursts would be produced by single ASCII characters from, and echoed to, computer terminals, when carried in individual packets. On the other hand, circuit-switched traffic and data file transfers are examples of streamed flow.

Consider the total bit rate, R, that results from traffic consisting of b, short bursts per second, and an aggregated stream flow of S bits per second. If the packet has h bits of header and x bits of DATA, then

$$R = [b(h + x)] + [(S/x)(h + x)]. \qquad (3)$$

This will be a minimum when

$$x = \sqrt{(S \cdot h)/b}. \qquad (4)$$

In a system serving a business, one might provide for a busy-hour voice traffic of 10 ccs (hundred call seconds) per telephone. In the switch, this will divide equally between the two memories. With 10,000 telephones, the aggregate stream S_v on each SM due to voice would then be

$$S_v = 10{,}000 \cdot 5 \cdot 64{,}000/36 = 89 \text{ Mb/s}.$$

A reasonable assumption for the present is that all other traffic would amount to 20 percent of the total, or in our example it would be a further 22 Mb/s.

For the sake of illustration, assume that the very short burst rate, b, is 20,000 packets per second. If each of these carries only one 8-bit byte, then the net traffic from them is 160 kb/s, a negligible amount within the assumed 22 M/bs. But the gross traffic may be much larger, depending on packet size. Hence, the decision for best size of DATA field, which, with the numbers already invoked, follows from eq. (4):

$$x = \sqrt{(111{,}000{,}000 \cdot 32)/20{,}000} = 421 \text{ bits}.$$

For x_{opt} to be less than 192 bits, decided by delay considerations, the very short burst traffic would have to be 4.8 times larger than was assumed. But the assumed rate is already large, and therefore it is unlikely that efficiency considerations would indicate a smaller packet than given from delay.

Given packets of 224 bits and the numbers cited above, the rate, R, in each memory follows from eq. (3) as 134 Mb/s. If 8-bit bytes are propagated in parallel, then the required clock rate is 16.75 MHz, a none too demanding frequency for present technology. The packet rate, which is of greater relevance to AU and CC speeds, would be 598 kHz.

A switch would be designed for a given ultimate size and given an appropriate clock rate from the start. But it would not be necessary to give it immediately the full complement of AUs, nor, indeed, full lengths of memories. AUs could be added without any disruption and memory sections with only a minor pause.

IV. RELIABILITY

Availability of communications services is extremely important and has prompted switch designers to adopt the very highest standards of reliability.[18,19] Thus, it is accepted practice to have two identical central controllers, one being a "hot" standby that can take over at any instant. This and other common practices would also apply to our switch. The features by which our switch is rendered most vulnerable in respect to reliability are its serial memories, which carry all messages and are accessed by all AUs. Below we consider the general question of disruptive impact by failures and suggest a measure for it. Then we introduce a fault detection and isolation scheme that would make the robustness of switching by serial memories with multiple read-and-write taps comparable to that of much more redundant architectures.

4.1 Failure impact

In switching equipment, including ours, failures are unequal in likelihood and in disruptive consequence. We introduce the notion of expected failure impact. Let π_{ik} be the probability that component C_i will fail during the course of one year; let the expected repair time for it be τ_{ik}; and the number of potential communication connections that are unavailable while C_i is in the failed state be ν_{ik}. We define U_i, the expected per annum failure impact (EPAFI) of C_i, as

$$U_i = \sum_k \pi_{ik} \cdot \tau_{ik} \cdot \nu_{ik}. \qquad (5)$$

We assume that failures are statistically independent and disregard the probability of another component failing during the repair time of an existing failure. EPAFI values then are additive, and U, with respect to an assembly of N components, is

$$U = \sum_{i=1}^{n} U_i. \tag{6}$$

We consider the expected failure of the SMs and all the AUs connected to them. Suppose that there are altogether N AUs, each with (1) a per annum rate π_1 of failing in a way that affects only one subscriber and takes time τ_1 to repair, and (2) a rate π_2, which disrupts communications on the memory past the failed AU and takes τ_2 to repair. Also, let the memories have a rate π_3 of failing at each of the $2N$ connecting points, with expected repair times τ_3.

With N mutually communicating AUs in the system, the number of potential two-way communication links is $N(N-1)/2$. If an AU failure of the first kind occurs, then $\nu_1 = (N-1)$ of these are disrupted. When the failure of the AU is of the second kind, or when a memory fails, then the number of disrupted links is much larger and depends on the actual location of the failure. One can calculate an average number on the assumption that all potential failure locations are equally likely. It is found to be approximately $(N^2)/3$. Hence the total expected impact due to failures of AUs and memory links is

$$U = [N\cdot(N-1)\cdot\mu_1] + [(N/3)\cdot N^2\cdot\mu_2] + [2\cdot(N/3)\cdot N^2\cdot\mu_3], \tag{7}$$

where $\pi_i\nu_i$ has been contracted to μ_i.

4.2 Failure detection and isolation

We propose to divide the memories into sections and have a fault detector at the end of each section. Further, each section would have a bypass and, in case of a detected failure, a switch would be actuated to pass on to the next section the data stream at the output of the bypass (Fig. 11). Thus, effectively, the consequence of the fault is isolated to one section. A possible realization of the switch is shown in Fig. 12.

A Fault Detector (FD) would compare the data streams at the outputs of the memory section and the bypass, and it would decide that failure has occurred when the evident modification to the stream in passage through the memory section violates existing constraints. The particular constraint of the several that exist in our case and which we use is the following: There may never be a change of any field that is already nonzero. Detecting any such illegal changes will catch failures both in AUs and the memories. The detection will, of course, rely on the output from the bypass being a flawless replica of what entered that section. If necessary, redundancies and error control could be implemented on the bypasses to make that more sure.